

ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD CARLOS III DE MADRID



PROYECTO FIN DE CARRERA

*BASES DE DATOS: SISTEMA DE GESTIÓN DE RUTAS
TURÍSTICAS*

INGENIERÍA TÉCNICA EN INFORMÁTICA DE
GESTIÓN

Autor: Gladys Alarza Blázquez

Tutor: Manuel Velasco de Diego

Septiembre 2009



AGRADECIMIENTOS

En primer lugar, me gustaría agradecer a mis padres todo el esfuerzo y apoyo que me han prestado siempre, ya que gracias a su aportación psicológica y económica he podido sacarme la carrera sin ninguna complicación.

A mis tíos por sus cuidados y su preocupación durante todos los años que he estado conviviendo con ellos. A mi hermano Fernando, por los ánimos y esperanzas que me ha dado.

A mi novio Mario por su apoyo y comprensión, tanto en los momentos buenos como en los momentos de bajón.

A mis abuelas, por estar continuamente pendiente y preocupadas por la evolución de mis estudios.

Por último, dar las gracias a todos aquellos que de alguna manera me han ayudado en la consecución de este proyecto.



ÍNDICE

AGRADECIMIENTOS	2
1. INTRODUCCION.....	4
1.1. DESCRIPCIÓN DEL PROBLEMA	4
2. ESTADO DE LA CUESTIÓN	5
2.1. INTRODUCCIÓN	5
2.2. CREACIÓN DE UNA BASE DE DATOS EN ACCESS 2003.....	7
2.2.1. ¿Cómo crear un fichero de base de datos?	7
2.2.2. Los menús de Access	10
2.2.3. Tablas.....	11
2.2.4. Los campos	18
2.2.5. Formularios.....	23
2.2.6. Filtros	29
2.2.7. Consultas	30
2.2.8. Relaciones	34
2.2.9. Informes.....	36
2.2.10. Macros	39
2.2.11. Módulos.	44
3. OBJETIVOS	47
4. MÉTODO DE DESARROLLO	49
4.1. Requisitos.....	49
4.1.1. Requisitos Hardware.....	50
4.1.2. Requisitos Software	50
4.1.3. Requisitos de Usuario.....	50
4.2. Dominio de la Base de Datos	50
4.3. Diseño de la Base de Datos	53
4.3.1. Modelo Conceptual: el modelo E/R	53
4.3.2. Modelo Lógico: el modelo Relacional.....	56
4.3.3. Modelo Físico.	57
4.4. Implementación de las tablas de la Base de Datos.....	61
4.5. Relaciones	66
4.6. Consultas de Bases de Datos	66
4.6.1. Consultas de Selección	67
4.7. Formularios.	83
5. EXPERIMENTACIÓN	107
5.1. CASOS DE PRUEBA DE INSERCIÓN DE DATOS	107
5.2. CASOS DE PRUEBA DE CONSULTAS	118
6. CONCLUSIONES	136
7. DESARROLLOS POSTERIORES	138
8. BIBLIOGRAFIA	140



1. INTRODUCCION

1.1. DESCRIPCIÓN DEL PROBLEMA

Se quiere desarrollar una base de datos que almacene rutas turísticas, para posteriormente trabajar con ella en diferentes aspectos. El escenario escogido para el desarrollo de la base de datos, es el mapa de la provincia de Cáceres. Este escenario recogerá las distintas partes de que se compone realmente, zonas, ciudades, rutas, carreteras, monumentos y alojamientos.

Uno de los trabajos posteriores que se realizarán sobre la base de datos, será la comprobación de la utilidad de los mecanismos de detección de errores cuando se introducen datos en la base de datos.

Otro de los trabajos posteriores, será un estudio de las consultas sobre bases de datos de los objetos y la implementación de algunas de ellas.

Se usará como sistema de coordenadas geográficas, los puntos cartesianos "X" e "Y" para posicionar las ciudades en el mapa.

En definitiva lo que se pretende es generar una base de datos, que nos sirva para desarrollar sobre ella, inserciones de datos y consultas que se probarán y usarán en una fase de experimentación.



2. ESTADO DE LA CUESTIÓN

2.1. INTRODUCCIÓN

Antes de proceder a explicar la herramienta utilizada para la realización del proyecto, es necesario tener claro todos los conceptos y definiciones de las bases de datos en general. Para ello, a continuación se muestra toda la información necesaria para que dichos datos sean comprendidos.

• ¿Qué es una base de datos?

Un gestor de base de datos es un programa que permite introducir y almacenar datos, ordenarlos y manipularlos. Organizarlos de manera significativa para que se pueda obtener información no visible como totales, tendencias o relaciones de otro tipo. Debe permitir en principio:

- Introducir datos
- Almacenar datos
- Recuperar datos y trabajar con ellos

Un programa de base de datos almacena la información que introducimos en forma de **tablas** como las que podemos ver, por ejemplo, en un listín telefónico:

Listín telefónico		
Nombre	Dirección	Teléfono
Cabrera Ortiz, Pedro	C/Mayor, 12	(948) 123457
García García, Ana	Avda. Arroyos, 54	(948) 559566
Santos Gemio, Luis	c/ Berruguete, 74	(948) 551234

Diagram labels: "Registro" points to a row; "Campo" points to a column; "Tabla" points to the entire table structure.

Figura 0: Ejemplo Listín Telefónico

En este listín nos interesa tener almacenados de modo ordenado los datos de una serie de personas. Para que aparezcan de modo claro los hemos



desglosado en tres apartados: Nombre, Dirección y Teléfono, haciendo que aparezca cada uno en una columna diferente. Así es mucho más sencillo encontrar la dirección de una persona buscando a partir de su nombre.

Aquí podemos ver cómo la información referida a una persona, "un dato", aparece en una fila de la tabla: a esto es a lo que se denomina **Registro** (El registro agrupa la información asociada a un elemento de un conjunto y está compuesto por campos). A cada una de las partes en las que hemos desglosado la información se le denomina Campo, y al conjunto formado por todos los registros, **Tabla** (conjunto de registros homogéneos con la misma estructura).

En el listín telefónico podemos ver que hay unos campos más importantes que otros: así el **Nombre** es fundamental para que el registro tenga sentido. Sería absurdo que apareciera una dirección en el listín sin ir acompañado de un nombre. Por este motivo se suelen denominar campos fundamentales a aquellos que definen al registro, y campos secundarios a los que lo complementan.



Figura 1: Esquema componentes base de datos

Hay dos tipos de bases de datos:

- Bases de datos planas. Es aquella que tan solo tiene una tabla, y en ella está almacenada toda la información.
- Bases de datos relacionales. Esta **base de datos relacional** estará formada por **tablas**, con la característica de que las mismas se relacionan entre sí mediante uno o más **campos**. Se puede decir que cada **objeto** de la realidad será una **tabla** en nuestra base de datos y que hay que buscar la manera de reflejar las **relaciones** entre ellas.

Por último antes de ver el programa en sí, es importante que se tenga claro que pasos hay que seguir al diseñar una base de datos:



- Es importante conocer exactamente para qué se quiere usar la base de datos, qué datos son los que interesan de los que existen en la realidad y qué información se necesitará extraer.
- Una vez que esto esté claro, se definen las Tablas que compondrán la base de datos. Dentro de cada tabla, se piensa qué campos serán necesarios. Conviene detenerse y definir correctamente la base de datos, ya que un mal diseño hará que el sistema sea lento y los resultados no sean los esperados

2.2. CREACIÓN DE UNA BASE DE DATOS EN ACCESS 2003.

2.2.1. ¿Cómo crear un fichero de base de datos?

Al iniciar el programa se presentará la siguiente ventana:



Figura 2: Crear un fichero de base de datos



Se selecciona Crear un archivo nuevo y se hace doble clic. Aparecerá otro cuadro de diálogo en el que se podrá elegir entre crear una Base de datos en blanco, una página de acceso a datos en blanco, abrir un archivo ya existente...



Figura 3: Crear fichero nuevo de base de datos

Para crear un archivo nuevo pulsaremos sobre Base de datos en blanco. Aparecerá otro cuadro de diálogo en el que se dará nombre a la base de datos que se va a crear. Se debe utilizar un nombre apropiado y relacionado con el contenido de la base de datos, para poder recuperarla con facilidad posteriormente.

Una vez creado el archivo, aparecerá la ventana de Access:

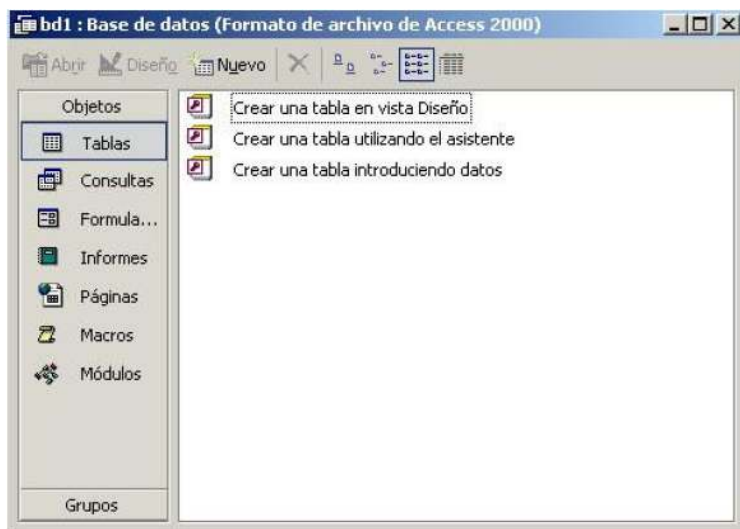



Figura 4: Guardar fichero nuevo de base de datos



Desde esta ventana se trabajan las bases de datos de Access. Seleccionando las pestañas se accede a los distintos elementos que componen una base de datos; tablas, consultas, formularios, informes, macros y módulos.

Para volver a esta ventana desde cualquier otra se pulsa el botón .

En el apartado de cada elemento de la base de datos se explicará como crearlo desde la ventana de cada uno. Pero desde la ventana de base de datos hay dos formas de crear cualquier elemento sin necesidad de seleccionar la pestaña: Tabla, Consulta....



Este botón cambia dependiendo del último objeto creado. Al pulsarlo se creará otro objeto similar al último creado. Si se quiere elegir otro objeto sólo hay que hacer clic sobre la flecha de la derecha. Se desplegará el menú de todos los elementos de Access:



Figura 4: Menú de objetos de la base de datos

En este menú se encuentran todos los elementos que componen Access. Para crear uno de ellos, basta con situar el ratón por encima de él y hacer clic.

Otra forma de crear un elemento de Access sin seleccionar la pestaña es a través del menú Insertar.



2.2.2. Los menús de Access

En la ventana de Access, como en todos los programas de Microsoft, hay un Menú y una Barra herramientas. La diferencia de Access es que tiene 22 barras de herramientas distintas. Las barras de herramientas y el menú varían según el elemento que esté seleccionado: tablas, consultas, formularios, informes...



Figura 5: Menú de Access

Nada más crear una base de datos, no todos los botones están activos. Irán variando según se vayan añadiendo elementos a la base de datos y según lo que se seleccione; tabla, consulta, formulario, informe, macros, módulos...

La barra de herramientas tiene los botones agrupados por la función que realizan, muchos de estos grupos se repiten en las diferentes barras. Estos son todos los botones que componen la barra de herramienta de la ventana general de Access. Según se vayan viendo los diferentes elementos de Access se verá que en sus barras aparecen muchos de estos elementos y otros nuevos:

	1º. Muestra las vistas disponibles 2º. Abre una base de datos ya existente 3º. Guarda la base de datos actual
	1º. Permite buscar archivos en el disco duro o en una unidad de red 2º. Imprime el elemento seleccionado 3º. Muestra la presentación preliminar de lo que se imprimirá
	1º Corrige la ortografía 2º Cortar 3º. Copiar 4º Pegar
	Deshacer: deshace la última acción realizada.

Figura 6: Características de los menús



	Estos son los vínculos que tiene Access con el Office: Con el MS Word para combinar y publicar (correspondencia) Con Excel para analizar los datos
	Estas opciones analizan los datos
	Editor de secuencias de comandos en Microsoft
	Código
	Muestra las propiedades del elemento que esté seleccionado, la fecha de creación, y la última modificación
	1º. Crea nuevos elementos de la base de datos
	1º. Muestra las relaciones existentes entre las tablas o ayuda a crearlas 2º Llama al ayudante de Office

Figura 6: Características de los menús

2.2.3. Tablas

Para empezar a trabajar con una base de datos primero es necesario crear las tablas. Dentro de cada una hay que definir los campos que contendrán la información.

1.1.1.1. ¿Cómo se crea una tabla?

Para crear una tabla se selecciona la pestaña



y

aparecerá la siguiente ventana:



Figura 7: Crear nueva tabla

Estas son las diferentes opciones que presenta Access para crear una tabla:

1. Vista Hoja de datos: crea una nueva tabla con formato de tabla. En la primera fila de la tabla aparecen los campos: Campo 1, Campo 2... sobre los cuales se escriben los nombres de los campos.

2. Vista Diseño: permite crear los campos manualmente y configurar el diseño de la tabla.

3. Asistente para tablas: el asistente pide las características de los campos y de la tabla y la genera automáticamente.

4. Importar tabla: esta opción permite importar datos de otra base de datos, que no necesariamente tiene que estar creada por Access.

5. Vincular tabla: crea vínculos entre las tablas importadas y las originales, las modificaciones que se efectúen en los datos se transmiten a aquéllas.

Cualquiera de estas opciones es válida para crear una base de datos. Pero las opciones a través de las cuales se crea personalmente una tabla son Vista Hoja de datos y Vista Diseño.

Con la opción Vista Diseño se crea la estructura para luego rellenar los datos en Vista Hoja de datos. Pero también se puede empezar directamente en Vista Hoja de datos introduciendo la información y Access crea la estructura automáticamente.



1.1.1.2. *Vistas de la tabla*

Las tablas se pueden ver desde dos vistas distintas, en cada una de ellas no sólo cambia el aspecto de la tabla, sino que además varían el menú y la barra de herramientas.

1.1.1.2.1. Vista hoja de datos

Con esta opción es más fácil entender cómo se almacena la información en una base de datos.

A continuación se va a crear la tabla Pacientes. Se selecciona Vista Hoja de datos y se pulsa Aceptar.



Figura 8: Vista hoja de datos

En esta tabla ya se puede empezar a introducir datos, aunque antes conviene dar nombre a los campos. Para esto se hace doble clic sobre Campo1 y se escribe el nombre que se le quiera dar al campo. En el caso de la tabla Pacientes el primer campo va a ser "Nombre". Los nombres de los campos tienen que cumplir unas normas. No pueden tener más de 64 caracteres, no pueden tener puntos, exclamaciones o corchetes.

Por otro lado no puede haber dos campos que se llamen de la misma manera.



Nombre	Campo2	Campo3	Campo4

Figura 9: Asignación de nombres a campos

Para que Access cree la estructura de la tabla basta con introducir los nombres de los campos y un registro.

Para escribir el primer registro se sitúa el ratón sobre el campo y se escribe el primer dato. Para pasar a la siguiente columna se puede usar el ratón o la tecla tabulador.

Siguiendo el mismo procedimiento, doble clic sobre la primera fila, se añaden los campos: apellidos, fecha de nacimiento y dirección.

Nombre	Apellidos	Fecha de naci	Dirección
Pepe	López	3-5-61	C/ Pio XII 4, 5D

Figura 10: Inserción de datos en tablas

Como aún no se han definido las características de los campos, es conveniente hacerlo antes de introducir más datos, esto se hace en el modo de vista diseño.

1.1.1.2.2. Vista diseño

Se elige Vista Diseño y se pulsa Aceptar. Access pedirá que se le dé un nombre a la tabla.

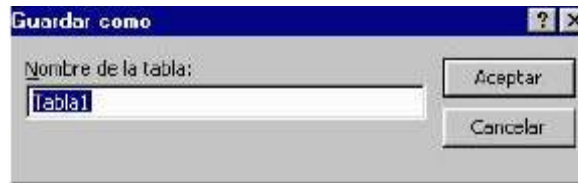


Figura 11: Guardar tablas

Se escribe el nombre: Pacientes en el recuadro y se pulsa Aceptar.

A continuación aparecerá otro mensaje comunicando que no se ha creado una clave principal.

Por el momento se pulsa No, ya que no se va a crear ahora, se verá más adelante en este manual.

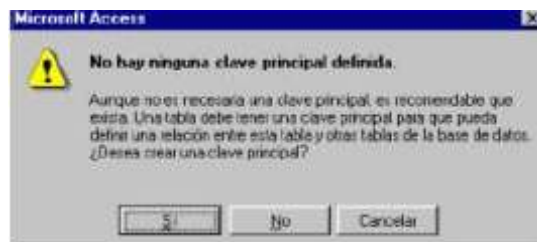


Figura 12: Error de no definición de clave principal

Después de pulsar No, se abrirá la ventana de diseño de la tabla ya creada con el tipo de dato definido por Access de forma automática.



Figura 13: Definición del tipo de datos de los campos



En la primera columna se ven los nombres de los campos, y en la segunda columna el tipo de datos que ha elegido automáticamente Access con los datos introducidos. Como se ve en cada columna se introducen diferentes características del campo:

- **Nombre del campo:** los nombres de los campos.
- **Tipo de datos:** texto, numérico, fecha/hora, contador, Si/No, memo, moneda, objeto OLE.
- **Descripción:** en esta columna se introduce la descripción del contenido del campo o su finalidad.
- **Propiedades de los campos:** estableciendo las propiedades de los campos se controla la apariencia de los datos, y se puede evitar que se introduzcan de modo incorrecto.

Para introducir el nombre se hace clic sobre la celda correspondiente y se teclea el nombre que se le vaya a dar al campo.

Para cambiar el tipo de dato, basta con situarse sobre el campo. En la celda de tipo de datos aparecerá una flecha. Al pulsar sobre ella se despliega un menú en el que se puede seleccionar el tipo de dato.

Nombre del campo	Tipo de datos
Nombre	Texto
Apellidos	Texto
Fecha de nacimiento	Texto
Dirección	Texto

Figura 14: Cambiar el tipo de datos de los campos

1.1.1.3. Clave principal

La clave principal suele ser uno o varios de los campos de la tabla. El contenido de este campo identifica cada registro del campo de manera única. De modo que no se podrán introducir dos registros iguales o almacenar valores nulos en los campos de la clave principal.

Para establecer un campo como clave principal se hace clic sobre él y en la barra de herramientas se pulsa el botón Establecer Clave Principal. También se puede realizar esta operación desde el Menú Edición, Clave Principal o pulsando con el botón derecho sobre el campo que queremos que sea la clave y aparecerá un menú desplegable y seleccionamos clave principal.

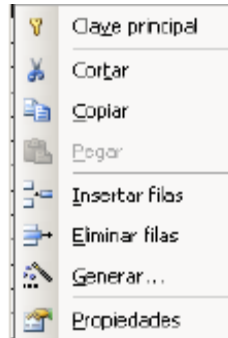


Figura 15: Establecer un campo como clave primaria

Al establecer un campo como clave principal aparecerá una llave a la izquierda del campo, tal y como muestra la imagen:

Nombre del campo	Tipo de datos	Descripción
 Código del paciente	Autonumérico	Código único de cada paciente
Nombre	Texto	Nombre del paciente
Apellidos	Texto	Primer y segundo apellido

Figura 16: Campo como clave primaria

No se tiene que definir obligatoriamente una clave principal, pero normalmente es conveniente hacerlo. Si no se establece la clave principal, al cerrar la tabla aparece un cuadro de diálogo pidiendo que se establezca:

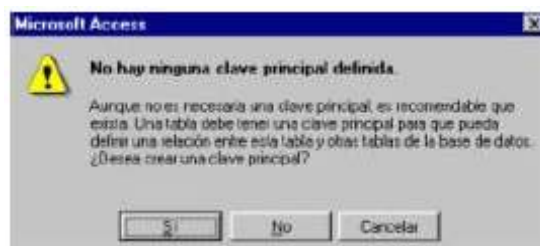


Figura 12: Error de no definición de clave principal

Si se elige la opción "Sí", Access creará automáticamente un campo Auto numérico que será la clave principal.




1.1.1.4. Abrir y trabajar una tabla

Para abrir una tabla de una base de datos ya existente, primero se tendrá que estar en esa base de datos.

Para abrir una base de datos, se selecciona Archivo, Abrir.

Aparecerá una ventana de diálogo con las diferentes bases de datos, se selecciona la que interese y se pulsa Abrir.

Una vez abierta la base de datos, siempre aparece la ventana de la base de datos, con múltiples pestañas. Se elige la pestaña  Tablas y se hace clic sobre la tabla que se desea abrir.

Se pulsa el botón Abrir, si se desea introducir datos, o Diseño si se desea añadir un campo o variar las propiedades de los campos.

Una vez se encuentra dentro de una de estas dos opciones siempre se tiene la oportunidad de cambiar pulsando el botón cambiar de vistas:



Figura 17: Vistas de una tabla

2.2.4. Los campos

Para crear los campos de una manera más completa es necesario profundizar en cada una de las características de un campo.

El nombre del campo debe ser descriptivo de la información que el campo va a contener para no crear confusión a la hora de trabajar con los datos.

Los campos se les deben asignar uno de los siguientes tipos de datos:

- **Texto:** almacena cadenas de caracteres, ya sean números (con los que no se vaya a realizar operaciones), letras o cualquier símbolo.
- **Numérico:** almacena números destinados a realizar operaciones. Hay cinco tamaños:
 - Byte: para almacenar el rango de 0 a 255



- Entero: para el rango entre -32768 y 32767
- Entero Largo: para el rango entre -2.147.483.648 y 2.147.483.647
- Simple: para números decimales entre el $-3,4 \times 10^{38}$ y el $3,4 \times 10^{38}$ con 7 decimales
- Doble: Doble para números entre el $-1,797 \times 10^{38}$ con 15 lugares decimales.
- **Fecha/hora:** fecha y hora general, fecha y hora larga, fecha y hora corta.
- **Autonumérico:** es un valor numérico que Access incrementa de modo automático cada vez que se añade un registro. No se puede modificar manualmente.
- **Si/No:** para almacenar datos que sólo tengan dos posibilidades: si-no, 0-1, verdadero-falso, blanco-negro...
- **Memo:** para almacenar texto largo, hasta de 64000 bytes.
- **Moneda:** para almacenar valores de moneda.
- **Objeto OLE:** son objetos tales como gráficos, texto, imágenes, creados en otras aplicaciones, que se han incrustado o vinculado.

1.1.1.5. Propiedades de los campos

Estableciendo las propiedades de los campos en Vista Diseño se controla la apariencia de los datos, y se puede evitar que se introduzcan de modo incorrecto.

- **Tamaño del campo**

Esta propiedad la pueden tener tanto los campos de tipo numérico como los de texto.

En los campos de texto el tamaño del campo indica el número de caracteres que puede contener. El valor máximo es de 255 caracteres y por defecto Access aplica 50. Se puede introducir un valor inferior, pero si ya se han introducido datos hay que tener mucho cuidado, ya que si se da un tamaño inferior al de algún campo los caracteres restantes se perderán.



En los de tipo numérico limita el rango de valores permitido y si el campo puede contener valores fraccionarios. El campo numérico puede tener estos tamaños: Byte, Entero, Entero Largo, Simple y Doble.

Configuración	Rango	Lugares Decimales	Lugar que ocupa
Byte	0 a 255	Ninguno	1 byte
Entero	-32.768 a +32.768	Ninguno	2 bytes
Entero Largo	-2.147.483.648 a + 2.147.483.647	Ninguno	4 bytes
Simple	$-3,4 \times 10^{38}$ a $3,4 \times 10^{38}$	7	4 bytes
Doble	$-1,797 \times 10^{308}$ a $1,797 \times 10^{308}$	15	8 bytes

Figura 18: Características de los tipos de datos

- **Formato**

Esta propiedad la pueden tener todos los campos menos los Objetos OLE.

El formato sólo afecta a la presentación de los datos, nunca al valor almacenado de una tabla.

Los números y las fechas se pueden presentar con diferentes formatos.

Los números pueden aparecer con separador de miles, con un símbolo de moneda o con un determinado número de decimales.

Si no se determina nada en esta propiedad Access utiliza el formato General y los datos aparecen tal y como se hayan introducido.

El formato que se especifique para un campo de una tabla será el que Access utilice para los formularios e informes basados en dicha tabla. Si se realiza un cambio de formato después de haber creado un formulario o un informe, este formato nuevo no le afectará.

Para garantizar la coherencia entre las distintas aplicaciones, Access utiliza los valores establecidos para los formatos de número y de fecha/hora en la sección internacional del Panel de Control de Windows.



- **Lugares decimales**

Esta propiedad sólo la tienen los campos de tipo numérico y de moneda. Determina el número de cifras decimales en la presentación de los campos.

Esta propiedad tiene dos configuraciones:

- De 0 a 15: aparecerán tantas cifras decimales como se indiquen sin tener en cuenta las que se especifiquen en el formato.
- Auto: aparecerá el número de cifras decimales predeterminadas para cada formato o la que este definida en la propiedad formato.

- **Máscara de entrada**

Esta propiedad la tienen los campos de texto, numérico, fecha/hora y de moneda. Obliga a que los datos introducidos en un campo se ajusten a una plantilla determinada.

Para crear una máscara hay un esquema de 3 partes separadas por ";":

- Presenta la máscara de entrada.
- Indica si los caracteres literales empleados en la máscara se almacenan o no en el campo junto con los datos. Se escribe 0 para que se guarden y 1 para que no se guarden.
- Especifica el carácter que debe aparecer en los espacios en blanco de la máscara. Access por defecto utiliza el subrayado.

- **Título**

Esta propiedad la tienen todos los tipos de campos.

Especifica la etiqueta que se utilizará en la presentación del campo cuando se crean tablas, formularios e informes.

- **Valor predeterminado**

Esta propiedad la tienen todos los campos menos los de tipo contador y Objeto OLE.



Introduce un valor por defecto en todos los campos. Este valor lo introduce el usuario cuando un campo va tener casi siempre el mismo valor. Cuando este valor varíe se puede modificar.

- **Regla de validación**

Esta propiedad se puede establecer en todos los campos menos en los de tipo Contador y Objeto OLE.

Se especifican las condiciones que deben cumplir los datos que se introduzcan, si los datos no cumplen las condiciones Access no admitirá ese dato.

Para introducir las condiciones que debe cumplir un campo se selecciona Regla de validación en las propiedades del campo. A la derecha del espacio en blanco hay unos puntos suspensivos. Se hace clic sobre ellos. Aparecerá una ventana para generar expresiones, en esta ventana habrá que especificar esas condiciones.

- **Texto de validación**

Cuando Access no admite un dato porque no cumple la regla de validación no aparece ningún mensaje que explique por qué no admite el dato a no ser que se utilice el texto de validación.

En esta propiedad se debe introducir cuál es la condición que debe cumplir el dato para que el usuario lo sepa.

- **Requerido**

Se aplica a todos los campos menos a los de tipo Contador.

Si se encuentra activado, no dejará que el usuario abandone un registro sin haberlo rellenado.

- **Permitir longitud cero**

Se aplica a los campos de tipo texto y memo.

Esta propiedad es útil para las consultas y expresiones, ya que los valores nulos se comportan de distinta forma.



Para introducir una cadena de longitud cero se teclea dos comillas dobles sin espacio entre ellas ("").

- **Indexado**

Se puede aplicar a todos los tipos de campo menos a Memo, Si/No, y Objeto OLE.

Esta propiedad crea un índice de ese campo. De modo que acelera las búsquedas de un registro por el contenido de ese campo. No conviene aplicarlo más que al campo por el que se vayan a realizar las búsquedas porque si no la actualización de los datos será muy lenta.

Sin duplicados: es una de las opciones de esta propiedad y significa que no puede haber dos campos con la misma clave. Con duplicados, hace que Access cree un índice normal con cada uno de los registros.

2.2.5. Formularios

La introducción de los datos directamente sobre las tablas es bastante incómoda. No sólo no se pueden ver todos los campos sin desplazarse con la barra de herramientas, sino que además los registros están uno encima de otro. Si se pierde la referencia del registro se pueden introducir datos que no correspondan a ese registro.

Los formularios permiten la introducción de datos en las tablas de una forma más sencilla y más limpia. En vez de introducir los datos directamente sobre la tabla, los datos se introducen en la tabla a través de los formularios. También puede usar un formulario como un panel de control que abra otros formularios e informes en la base de datos, o como un cuadro de diálogo personalizado que acepte los datos del usuario y realiza la acción basada en la entrada.

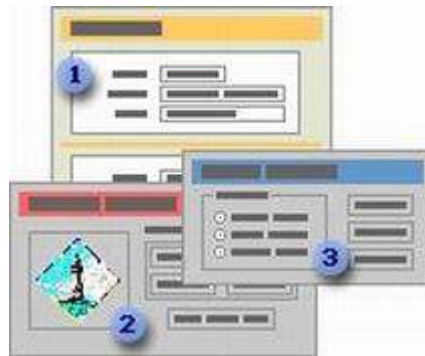


Figura 19: Dibujos tipos de formularios

Hay diferentes formatos de formularios, en alguno de ellos los registros se pueden ver de forma aislada, en otros, todos los registros juntos, o también se puede diseñar un formulario con un formato totalmente personalizado.

En una base de datos se puede crear más de un formulario basado en una misma tabla. Un formulario puede tomar varios campos de una tabla o todos, incluso puede tomar campos de diferentes tablas.

Las tablas almacenan la información, los formularios se encargan de recogerla. Para que las tablas puedan incluir los datos de los formularios deben estar cerradas. Al cerrar o guardar los formularios, las tablas se actualizan.

Pedidos : Tabl 1

Id. de pedido	Id. de cliente	Fecha de entrega	Empleado
11022	HANAF	09-May	Dodsworth, Anne
11023	BSBE V	25-May	Davolio, Nancy

FACTURA

Pedidos : For12ario

B's Beverages
Fanutleroy Circus
Londres
EC25NT
UK

Imprimir factura

Vendedor: Davolio, Nancy

Id de pedido: 11023 **Fecha de entrega:** 10-May

Producto	Precio unidad	Cantidad	Precio con descuento
Chai	18.00 \$	10	180.00 \$
Ipoh Coffee	46.00 \$	30	1380.00 \$

Figura 20: Obtención de campos de tablas en formularios

La mayoría de los formularios están enlazados a una o varias tablas y consultas (consulta: pregunta sobre los datos almacenados en las tablas o



solicitud para llevar a cabo una acción en los datos. Una consulta puede unir datos de varias tablas para servir como origen de datos de un formulario, informe o página de acceso a datos) de la base de datos. El origen de registros (origen de registros: origen de datos subyacente de un formulario, informe o página de acceso a datos. En una base de datos de Access, puede ser una tabla, una consulta o una instrucción SQL. En un proyecto de Access, puede ser una tabla, una vista, una instrucción SQL o un procedimiento almacenado) de un formulario hace referencia a los campos de las tablas y consultas base. No es necesario que un formulario contenga todos los campos de cada una de las tablas o consultas en las que está basado.

Un formulario dependiente almacena o recupera los datos del origen de registros base. Otra información del formulario, con el título, la fecha y el número de página, se almacena en el diseño del formulario.

- Los elementos gráficos como, por ejemplo, líneas y rectángulos, se almacenan en el diseño del formulario.
- Los datos proceden de los campos del origen de registros base.
- Un cálculo procede de una expresión almacenada en el diseño del informe.
- El texto descriptivo se almacena en el diseño del formulario.

El vínculo entre un formulario y su origen de registros se crea mediante los objetos gráficos denominados controles (control: objeto de interfaz gráfica para el usuario, como un cuadro de texto, una casilla de verificación, una barra de desplazamiento o un botón de comando, que permite a los usuarios controlar el programa. Utilice los controles para mostrar datos y opciones, realizar una opción o facilitar la lectura de la interfaz). El tipo de control más comúnmente utilizado para mostrar y escribir datos es un cuadro de texto.

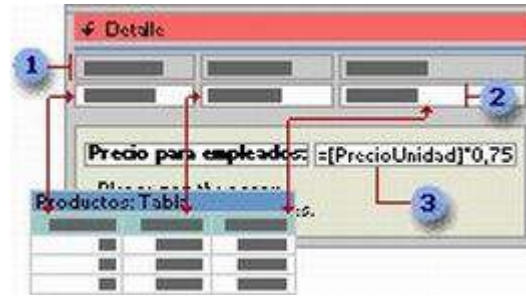


Figura 21: Actualizar los datos del formulario en las tablas

- Las etiquetas muestran texto descriptivo.
- Los cuadros de texto muestran los datos de la tabla Productos e introducen datos en dicha tabla.
- Un cuadro de texto utiliza una expresión para calcular un total.

También puede abrir un formulario en la vista Tabla dinámica o Gráfico dinámico para analizar los datos. En dichas vistas, puede cambiar dinámicamente el diseño de un formulario para presentar los datos de diferentes maneras. Puede reorganizar los encabezados de filas, encabezados de columnas y campos de filtro hasta que obtenga el diseño deseado. Cada vez que modifica el diseño, el formulario actualiza inmediatamente los datos a partir de la nueva organización.

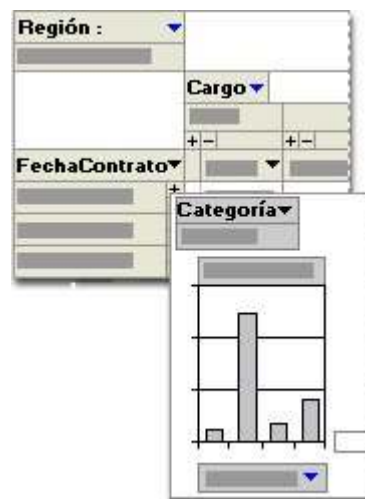


Figura 22: Vista formulario, tabla dinámica



- En la vista Tabla dinámica, puede ver los datos de detalles o datos resumidos organizando los campos en las áreas de filtros, filas, columnas y detalles.
- En la vista Gráfico dinámico, puede mostrar visualmente los datos al seleccionar un tipo de gráfico y ver los datos organizando los campos en las áreas de filtros, series, categorías y datos.

❖ **Creando un formulario**

Para crear rápidamente un formulario, utilice el comando Autoformato o un Asistente. La función Autoformulario crea un formulario que muestra todos los campos y registros de la tabla o consulta base. El asistente hace preguntas y crea un formulario basándose en las respuestas que obtiene. Después, podrá personalizar el formulario en la vista Diseño (vista Diseño: ventana que muestra el diseño de estos objetos de base de datos: tablas, consultas, formularios, informes, macros y páginas de acceso a datos. En la vista Diseño, puede crear objetos de base de datos nuevos y modificar el diseño de otros existentes).

En la vista Diseño, puede personalizar un formulario de las siguientes maneras:

- Origen del registro: puede cambiar las tablas y consultas en las que está basado un formulario.
- Controlar y ayudar al usuario: puede establecer las propiedades del formulario para permitir o impedir a los usuarios que agreguen, eliminen o modifiquen los registros mostrados en un formulario. También puede agregar Ayuda personalizada a un formulario para ayudar a los usuarios a utilizarlo.
- Ventana Formulario: puede agregar o quitar los botones Maximizar y Minimizar, cambiar los menús contextuales y otros elementos de la ventana Formulario.
- Secciones: se puede agregar, quitar, ocultar o cambiar de tamaño los encabezados, pies y las secciones de detalles de un informe. También



se puede establecer propiedades de sección para personalizar la forma en que se imprimirá el contenido de una sección.

- **Controles:** puede mover, cambiar el tamaño o establecer las propiedades de fuente de un control. También puede agregar controles para mostrar los valores calculados, totales, la fecha y hora actuales, así como otro tipo de información que sea de utilidad en un formulario. Los controles pueden ser dependientes, independientes o calculados.
 - **Control dependiente:** está unido a un campo de una tabla o consulta. Los controles dependientes se utilizan para mostrar, introducir y actualizar valores de los campos de la base de datos.
 - **Control independiente:** no tiene un origen en una tabla o consulta. Los controles independientes se pueden utilizar para mostrar información, líneas, rectángulos e imágenes, independientemente de que estos existan en la tabla.
 - **Control calculado:** el origen de los datos es una expresión, no un campo. Una expresión es una combinación de operadores ("=","+", "-", "*" y "/"), nombres de controles, nombres de campos, funciones que devuelven un solo valor y valores constantes.

A los controles se tiene acceso a través del cuadro de herramientas de la Vista diseño. Estos son los diferentes tipos de controles:



Cuadro de texto		Sirven para mostrar o introducir datos
Etiqueta		Crea una etiqueta
Grupo de opciones		Formado por un grupo de casillas de verificación o botones de opción. Sólo permite que se active una opción.
Botón de opción		Para valores Si/No. Se puede utilizar dentro de un grupo de opciones.
Casilla de verificación		Para valores Si/No. Se puede utilizar dentro de un grupo de opciones.
Botón de alternar		Para valores Si/No. No puede utilizarse dentro de un grupo de opciones.
Cuadro combinado		Permite seleccionar un elemento de una lista o escribir el dato directamente.
Cuadro de lista		Permite seleccionar un elemento de una lista.
Botón de comando		Inserta un botón que al ser pulsado ejecuta instrucciones.
Imagen		Inserta un marco para incluir una imagen. No es un objeto OLE. No se edita.
Marco de objeto dependiente		Inserta un marco para incluir un objeto OLE que depende del valor de un campo.
Marco de objeto independiente		Inserta un marco para incluir un objeto OLE que no depende del contenido de un campo.
Subformulario/subinforme		Permite introducir un formulario dentro de otro.
Salto de página		Cuando el formulario tiene mas de una página, así se indica dónde empieza cada una.
Línea		Inserta una línea en el formulario.
Rectángulo		Inserta un rectángulo.

Figura 23: Controles de formularios

2.2.6. Filtros

Un filtro es una selección de algunos elementos de una tabla o de un formulario. Aplicando un filtro es posible ver dentro de una tabla sólo aquellos registros que cumplen unas condiciones.

El resto de los registros no desaparecen, sólo están ocultos.

Access tiene tres tipos de filtros:

- Filtro por selección Este filtro se aplica seleccionando dentro de la tabla el elemento que va a ser la condición de filtrado.
- Filtro por formulario Este filtro es de gran utilidad para tablas grandes y con muchos registros.
- Filtro u orden avanzado Este filtro permite determinar con mayor minucia las características del filtro. Es muy similar a una consulta, y se almacena en Consultas




Para aplicar un filtro se selecciona el menú Registros|Filtro:



Figura 24: Menú Registros/Filtros

O bien se utilizan los botones de la barra de herramientas:



Para aplicar o desactivar un filtro se puede utilizar el botón . Aplicará el último filtro utilizado y desactivará el filtro que en ese momento esté activado.

2.2.7. Consultas

La consulta es una solicitud de información a la base de datos. Los datos mostrados pueden proceder de una sola tabla o de un conjunto de tablas. El resultado de la consulta es la "Hoja de respuestas dinámica"; en esta hoja se encuentra la información extraída de las tablas. Pero la información no se guarda en la hoja de respuestas, sino que sigue estando almacenada en las tablas.

En determinados tipos de consulta se puede modificar la información de las tablas, pero la consulta sigue siendo una forma de acceder a la tabla, no un objeto que almacene información.

La consulta muestra lo que la tabla almacena según los criterios solicitados.

La consulta es un filtro avanzado, y funciona prácticamente de la misma forma.

Lo único que los diferencia es que los filtros sólo se pueden activar desde una tabla o desde un formulario.



- **¿Qué puede hacer una consulta?**

- **Elegir tablas:** las consultas se pueden realizar sobre una sola tabla o sobre todas las tablas creadas en esa base de datos. De esta forma las combinaciones posibles para obtener información son muchas.
- **Modificar los datos de las tablas:** aunque las consultas no son tablas, dan acceso a ellas, y permite modificar, eliminar o añadir registros nuevos. También se puede utilizar una consulta para crear una nueva tabla que contenga registros de otra tabla o de varias tablas.
- **Elegir uno o varios campos:** al crear una consulta es posible especificar qué campo se desea ver.
- **Seleccionar registros:** una consulta se puede concretar hasta el punto de ver sólo un registro.
- **Realizar cálculos:** se pueden realizar cálculos con los campos mostrados en la consulta. Por ejemplo contar el número de registros seleccionados o acumular totales. Se crearán campos nuevos: campos calculados que contendrán el resultado del cálculo.
- **Para crear nuevos formularios, informes o consultas:** partiendo de los datos obtenidos en una consulta se pueden crear nuevos elementos.

- **Tipos de Consultas**

- **Consulta de selección.** Es la más sencilla, se indican unos criterios para ver sólo lo que interesa de una tabla. Los datos aparecen en la Hoja de respuestas dinámicas, esta parece una tabla pero no lo es, sólo muestra los datos de una tabla o de varias tablas según los criterios de la consulta. Aunque la hoja de respuestas dinámica no es una tabla se pueden introducir datos en las tablas a través de ella.
- **Consulta de tablas de referencias cruzadas.** Presenta los datos con títulos en las filas y en las columnas; la apariencia es la de una



hoja de cálculo. De esta forma se resume en muy poco espacio mucha información de una forma muy clara.

- **Consulta de parámetros.** Cuando se ejecuta, muestra un cuadro de diálogo que solicita la información para recuperar registros o un valor que desea insertar en un campo. Se puede diseñar la consulta para que solicite más de un dato, por ejemplo, entre dos fechas.
- **Consulta de acciones.** Una consulta de acción es una consulta que realiza cambios o desplazamientos de muchos registros en una sola operación. Hay cuatro tipos de consultas de acción:
 - **Consulta de eliminación:** Elimina un grupo de registros de una o más tablas. Por ejemplo, puede utilizar una consulta de eliminación para quitar productos que ya no se fabrican o de los que no hay pedidos. Con las consultas de eliminación, siempre se eliminan registros enteros, no sólo campos seleccionados dentro de los registros.
 - **Consulta de actualización:** Realiza cambios globales en un grupo de registros de una o más tablas. Por ejemplo, puede aumentar los precios un 10 por ciento para todos los productos lácteos o bien puede aumentar los sueldos un 5 por ciento al personal de una determinada categoría. Con una consulta de actualización, puede cambiar los datos de las tablas existentes.
 - **Consulta de datos anexados:** Agrega un grupo de registros de una o más tablas al final de una o más tablas. Por ejemplo, supongamos que consigue nuevos clientes y una base de datos que contiene una tabla con información acerca de estos clientes. Para evitar tener que escribir toda esta información en la base de datos, desea anexarla a la tabla Clientes.
 - **Consulta de creación de tabla:** Crea una tabla nueva a partir de la totalidad o una parte de los datos de una o más tablas. Las consultas de creación de tabla son útiles para crear una tabla que se desee exportar a otra base de datos de Microsoft Access o una tabla histórica que contenga registros antiguos.



- **Consulta de paso a través o consulta SQL:** Envía instrucciones a una base de datos SQL.
- **¿Cómo crear una consulta?**



Para crear una consulta hay que seleccionar la pestaña  Consultas de la ventana de la base de datos. Si no se está en la ventana Base de datos, para volver a ella se pulsa el botón base de datos . En la ventana Consultas puede aparecer algún elemento creado si se ha guardado algún filtro:



Figura 25: Menú Crear una consulta en vista Diseño

Para crear una consulta nueva se pulsa el botón Nuevo. Aparecerá esta ventana:

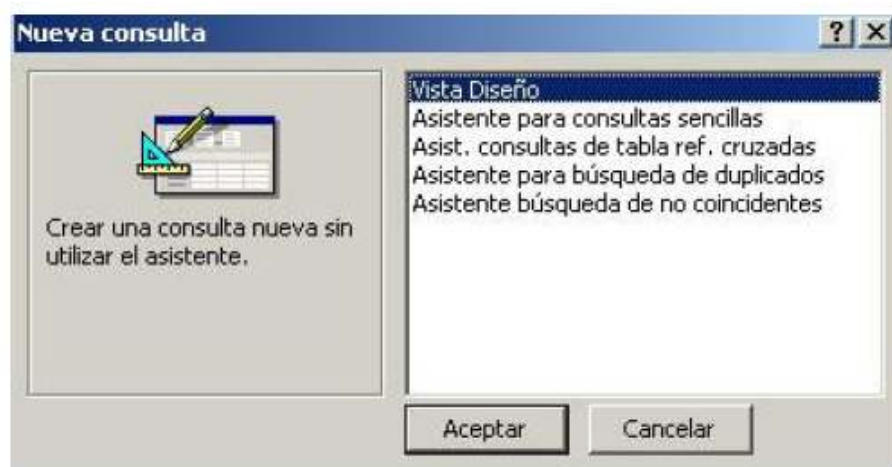


Figura 26: Menú Crear una nueva consulta



Estas son las opciones que ofrece Access para crear una consulta:

- **Vista diseño:** permite realizar una consulta sin la ayuda del asistente.
- **Asistente para consultas sencillas:** crea una consulta automáticamente, según los campos seleccionados.
- **Asistente para consultas de referencias cruzadas:** crea una consulta que muestra los datos con un formato compacto, parecido al de una hoja de cálculo.
- **Asistente para consultas destinadas a buscar duplicados:** crea una consulta en la que se buscan registros con valores duplicados en un campo.
- **Asistentes para consultas destinados a buscar no-coincidentes:** crea una consulta que busca registros que no tienen registros relacionados en otra tabla

2.2.8. Relaciones

La diferencia de las bases de datos relacionales con respecto a una base de datos plana consiste en que los datos sólo se introducen una sola vez en una tabla, pero gracias a las relaciones pueden aparecer en las tablas que se quiera. Cualquier modificación sólo hay que realizarla una sola vez y automáticamente se realizará en todas las demás tablas. De este modo se ahorra mucho tiempo, espacio y exactitud en los datos que siempre estarán actualizados independientemente de la tabla en la que estemos.

- **Tipos de Relaciones**
 - **Relación uno a uno.** Cada registro de la tabla A se relaciona sólo con un registro de una tabla B y cada registro de la tabla B se relaciona sólo con un registro de la tabla A. Relaciones de este tipo se almacenan guardando en la tabla el identificador de la otra tabla con la que mantiene la relación.

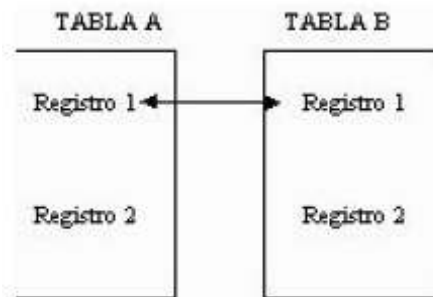


Figura 27: Relación 1 a 1

- **Relación uno a varios.** Cada registro de la tabla A está relacionado con varios registros de la tabla B y cada registro de la tabla B está relacionado con un sólo un registro de la tabla A. Aplicando esto al siguiente ejemplo de la figura, una relación de este tipo se daría entre la tabla pacientes y la tabla médicos, ya que el mismo médico se hará cargo de varios pacientes. Un solo registro de la tabla de médicos se relaciona con varios registros de la tabla de pacientes. Pero un registro de la tabla de pacientes sólo se relaciona con un registro de la tabla médicos.

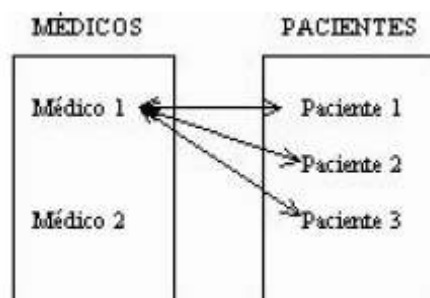


Figura 28: Relación 1 a N

- **Relación varios a varios.** Cada registro de la tabla A puede estar relacionado con más de un registro de la tabla B y cada registro de la tabla B puede estar relacionado con más de un registro de la tabla A. Si existiera una base de datos con dos tablas: médicos y pacientes, con una relación directa entre ellos, un médico atendería muchos pacientes



y un mismo paciente podría ser atendido por varios médicos. Varios registros de la tabla de médicos se relacionarían con varios registros de la tabla de pacientes. Relaciones de este tipo se almacenan creando una tabla especial donde se colocan los identificadores de cada tabla y otros campos que puedan ser de utilidad, por ejemplo la fecha, la hora, comentarios acerca de la visita médica, etc. En el siguiente ejemplo esta tabla especial es la tabla Visitas, donde aparecen tanto el código del médico como el del paciente.

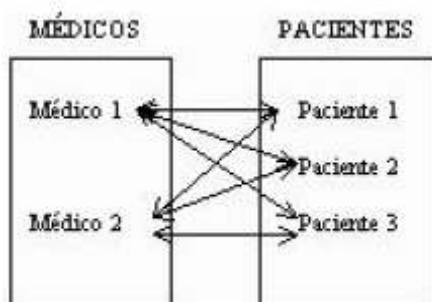


Figura 29: Relación N a N

2.2.9. Informes

Un informe es un objeto de Access. Los informes no guardan información, sólo son una presentación gráfica de los datos contenidos en tablas o los hallados en consultas. Esta presentación gráfica está orientada a la impresión de los datos. El diseño puede ser en columnas o en etiquetas, dependiendo de la utilidad que se le vaya a dar al impreso.

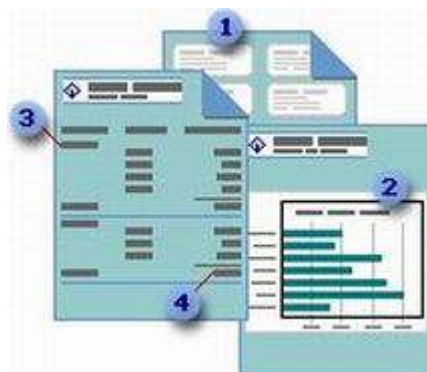


Figura 30: Esquema informes



- Utilice un informe para crear etiquetas postales.
- Utilice un informe para mostrar totales en un gráfico.
- Utilice un informe para calcular totales.
- Agrupar registros en categorías

• Crear un informe

Dentro de la ventana de la base de datos se selecciona la pestaña de Informes. Para crear un informe nuevo se hace clic sobre el botón Nuevo. Para modificar el diseño de un informe ya creado se selecciona Diseño. Para ver cómo se imprimirá el informe se selecciona Vista Previa.

Tras seleccionar Nuevo aparecerá la ventana dónde se indican los diferentes modos de creación de un informe:

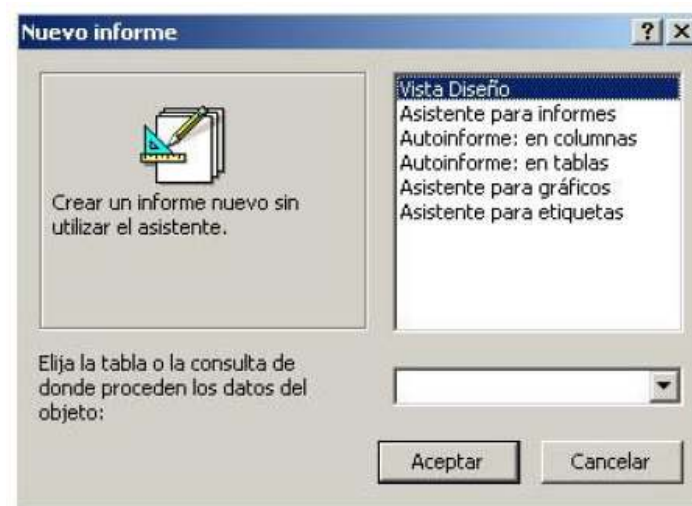


Figura 31: Menú Crear nuevo informe

- **Vista Diseño:** permite crear manualmente un informe sin ayuda de los asistentes.
- **Asistente para informes:** el asistente guía al usuario para la creación de varios modelos de informes.
- **Autoinforme columnas:** es una forma automática de crear un informe en columnas. Sólo hay que seleccionar la tabla o la consulta de la cual van a extraerse los datos. Los nombres de los campos aparecen en la



columna izquierda y a la derecha el dato. En cada registro vuelven a aparecer los nombres de los campos.

Pacientes

Código del paciente: 26

Nombre: Adnan

Apellidos: Ocaña

Sexo: Hombre

Fecha de nacimiento: 4/03/75

Edad: 24

DNI: 1145678

Dirección: C/ Encarnación 234

Población: Madrid

Provincia: Madrid

Teléfono: 897634

Código del paciente: 27

Nombre: Ocaso

Apellidos: Martínez

Figura 32: Ejemplo informe

- **Autoinforme tabular:** esta opción crea de forma automáticamente un informe. El diseño es en filas y columnas, como en una hoja de cálculo. En la primera fila aparecen los nombres de los campos y en el resto de las filas los datos.

Pacientes

Cód. pac.	Nombre	Apellidos	Sexo	Fecha de n.	Edad	DNI	Dirección	Población	Provincia	Teléfono
1	Paco	López	Hombre	1985-05	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
2	Ana	Ocaña	Mujer	1990-02	30	3145678	C/ Encarnación 234	Madrid	Madrid	897634
3	Juan Luis	Roberto	Hombre	1980-02	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
4	Antonio	Jiménez	Hombre	1980-05	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
5	Adnan	Martínez	Hombre	1985-03	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
6	Pablo	Rodríguez	Hombre	1980-08	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
7	Juan	Martínez	Hombre	1980-02	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
8	Stefano	Roberto	Mujer	1980-08	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
9	Stefano	Pérez	Mujer	1980-02	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
10	Yamir	Rodríguez	Mujer	1980-02	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
11	Diana	Ocaña	Mujer	1980-05	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
12	Osvaldo	Torres	Hombre	1980-01	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
13	Cooper	Daniela	Hombre	1980-02	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634
14	Leo	Calderón	Hombre	1980-05	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
15	María	Torres	Mujer	1980-01	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
16	Antonio	Ocaso	Hombre	1980-05	40	3145678	C/ Encarnación 234	Madrid	Madrid	897634
17	Pablo	Jiménez	Hombre	1980-08	35	3145678	C/ Encarnación 234	Madrid	Madrid	897634

selecciona 21 de abril de 1999

Página 1 de 2

Figura 33: Ejemplo informe tabulado

- **Asistente para gráficos:** es un asistente que crea informes gráficos. El asistente ayuda a elegir los campos a incluir en el gráfico y el tipo de



gráfico que se quiere. Con este asistente se seleccionarán campos que se desea que pasen a formar parte de un gráfico donde, por ejemplo, se pueden hacer comparaciones entre datos visualmente.



Figura 34: Asistente para gráficos de informes

- **Asistente para etiquetas:** ayuda a la creación de informes tipo etiquetas. Con este formato se pueden confeccionar etiquetas postales. El asistente ayuda a la elección de los campos a incluir en la etiqueta y al diseño de la misma.



Figura 35: Asistente para etiquetas de informes

2.2.10. Macros

Una macro es un conjunto de una o más acciones, que cada una realiza una operación determinada, tal como abrir un formulario o imprimir un informe. Las macros pueden ayudar a automatizar las tareas comunes. Por ejemplo, puede



ejecutar una macro que imprima un informe cuando el usuario haga clic en un botón de comando.



Figura 36: Menú Macro

Cuando se crea una macro, las acciones que se desea realizar se escriben en esta parte de la ventana Macro. En esta parte de la ventana se puede especificar los argumentos de una acción.

Una macro puede ser una macro compuesta de una secuencia de acciones, o puede ser un grupo de macros. También se puede usar una expresión condicional (expresión condicional: expresión que se evalúa y compara con un valor, por ejemplo, las instrucciones If...Then y Select Case. Si se cumple la condición, se llevan a cabo una o más operaciones. Si no se cumple, se omite la operación) para determinar si se llevará a cabo una acción en algunos casos cuando se ejecute la macro.

La siguiente macro está compuesta de una serie de acciones. Microsoft Access lleva a cabo estas acciones cada vez que se ejecuta la macro. Para ejecutar esta macro se hace referencia al nombre de la macro Revisar Productos.



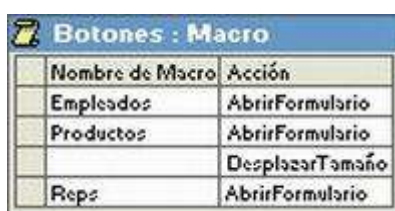
Figura 37: Ejemplo Acciones de una Macro



- **¿Qué es un grupo de macros?**

Cuando se tienen muchas macros, el reunir las que estén relacionadas en grupos de macros puede simplificar la administración de la base de datos.

Por ejemplo, el siguiente grupo de macros, denominado Botones, está compuesto de tres macros relacionadas: Empleados, Productos, y Representantes. Cada macro lleva a cabo la acción AbrirFormulario, y la macro Productos lleva también a cabo la acción DesplazarTamaño.



Nombre de Macro	Acción
Empleados	AbrirFormulario
Productos	AbrirFormulario
	DesplazarTamaño
Reps	AbrirFormulario

Figura 37: Ejemplo Acciones de una Macro

El nombre en la columna Nombre de macro identifica a cada macro. Cuando se ejecuta una macro en un grupo de macros, Microsoft Access lleva a cabo la acción de la columna Acción y cualquier otra acción que siga inmediatamente con una columna Nombre de macro en blanco.

Para ejecutar una macro de un grupo de macros en un evento o procedimiento de evento, escriba el nombre del grupo de macros seguido de un punto y del nombre de la macro. En el ejemplo anterior, para hacer referencia a la macro Empleados en el grupo de macros Botones, escribiría Botones.Empleados.

- **Acciones condicionales**

En algunos casos, puede desear llevar a cabo una acción o serie de acciones en una macro solamente si se cumple una condición concreta. Por ejemplo, si está utilizando una macro para validar los datos de un formulario, puede desear presentar un mensaje en respuesta a un conjunto de valores introducidos en un registro y otro mensaje en respuesta a otro conjunto diferente de valores. En



casos como éstos, puede utilizar las condiciones para controlar el flujo de la macro.

Una condición es una expresión lógica que se evalúa como Verdadero/Falso o Sí/No. La macro sigue trayectorias diferentes dependiendo de si la condición es verdadera o falsa.

Cuando ejecuta la macro, Microsoft Access evalúa la primera expresión condicional. Si la condición es verdadera, Microsoft Access lleva a cabo la acción en esa fila y cualquiera de las acciones inmediatamente posteriores que vayan precedidas por puntos suspensivos (...) en la columna condición.

Microsoft Access ejecuta entonces cualquier acción adicional en la macro que tenga una columna de Condición en blanco, hasta que alcance otra expresión, un nombre de macro o el final de la macro.

Si la condición es falsa, Microsoft Access omite la acción y cualquier otra acción inmediatamente posterior que vaya precedida por puntos suspensivos en la columna Condición. Después, se desplaza a la siguiente fila de acción que contenga otra condición o una columna Condición en blanco.

La siguiente macro ejecuta las acciones CuadroMsj y DetenerMacro sólo cuando la expresión en la columna Condición sea verdadera (cuando hay un valor Nulo en el campo IdProveedor).



Condición	Acción
EsNulo([IdProveedor])	CuadroMsj
	DetenerMacro

Figura 37: Ejemplo Acciones de una Macro

- **Argumentos de acción**

Los argumentos de acción son datos adicionales necesarios para algunas acciones de macro, por ejemplo, el objeto que se verá afectado por la acción o



las condiciones especiales en las que la acción se ejecuta. Después de agregar una acción a una macro, establezca los argumentos de la acción en la parte inferior de la ventana Macro (ventana Macro: ventana en la que se crean y modifican las macros.). Estos argumentos dan a Microsoft Access información adicional sobre cómo llevar a cabo la acción.

Sugerencias para establecer argumentos de acción:

- En general, es una buena idea establecer los argumentos de la acción en el orden en el que están listados, porque las opciones de un argumento pueden determinar las opciones de los argumentos que le siguen.
- Si agrega una acción a su macro arrastrando un objeto de base de datos (objetos de base de datos: una base de datos de Access contiene objetos como tablas, consultas, formularios, informes, páginas, macros y módulos. Un proyecto de Access contiene objetos como formularios, informes, páginas, macros y módulos.) desde la ventana Base de datos (ventana Base de datos: ventana que aparece cuando se abre una base de datos o un proyecto de Access. Muestra métodos abreviados para crear objetos de base de datos nuevos y abrir otros existentes.), Microsoft Access establece de forma automática los argumentos apropiados para esa acción.
- Si una acción tiene un argumento que requiere el nombre de un objeto de la base de datos, puede establecer de forma automática el argumento y el tipo de objeto correspondiente arrastrando el objeto desde la ventana Base de datos al cuadro del argumento.
- Puede utilizar una expresión (expresión: cualquier combinación de operadores matemáticos o lógicos, constantes, funciones y nombres de campos, controles y propiedades que evalúa a un solo valor. Las expresiones puede realizar cálculos, manipular caracteres o probar datos) precedida de un signo igual (=) para establecer muchos argumentos de acción.



2.2.11. Módulos.

Un módulo es básicamente un conjunto de declaraciones, instrucciones y procedimientos que se almacenan en una unidad con nombre para organizar el código de Microsoft Visual Basic. Microsoft Access tiene dos tipos de módulos: módulos estándar y módulos de clase.

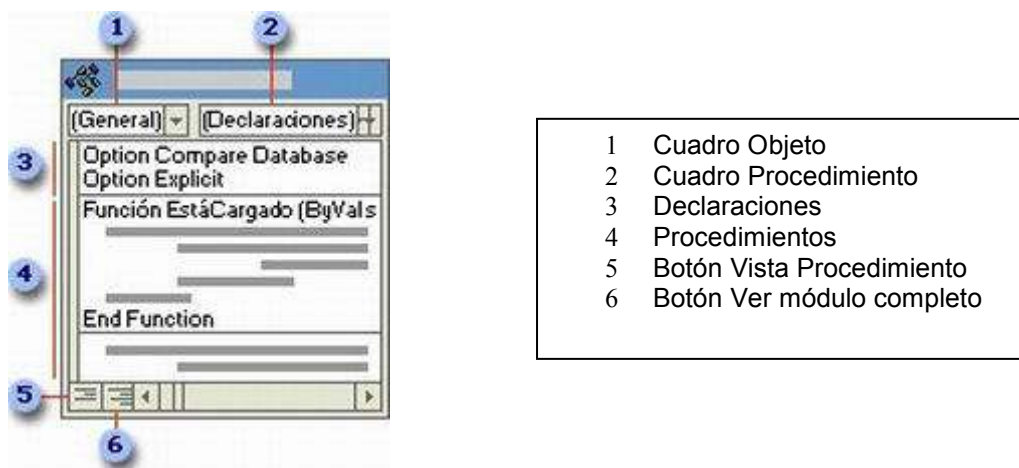


Figura 38: Elementos de un módulo

- **Módulos de clases**

Los módulos de formularios (módulo de formulario: módulo que incluye código para todos los procedimientos de evento desencadenados por los eventos que suceden en un formulario específico o en sus controles.) y los módulos de informes (módulo de informe: módulo que incluye código para todos los procedimientos de evento desencadenados por los eventos que tienen lugar en un informe específico o en sus controles) son módulos de clases que están asociados con un formulario o informe determinado. Los módulos de formularios y de informes contienen a menudo procedimientos de evento (procedimiento de evento: procedimiento que se ejecuta automáticamente en respuesta a un evento iniciado por un usuario) que se ejecutan en respuesta a un evento en un formulario o informe. Puede usar los procedimientos de evento para controlar el comportamiento de los formularios



e informes, y su respuesta a acciones de los usuarios, como hacer clic en un botón de comando.

Cuando se crea el primer procedimiento de evento para un formulario o informe, Microsoft Access crea automáticamente un módulo de formulario o un módulo de informe.

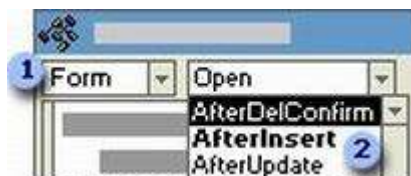


Figura 39: Menú de un módulo

Para ver o agregar un procedimiento de formulario, elija uno de los objetos del formulario en el cuadro Objeto y a continuación, seleccione un evento en el cuadro Procedimiento. Los nombres de eventos que ya tienen procedimientos aparecen en negrita.

Los procedimientos en los módulos de formularios y de informes pueden llamar a procedimientos que haya agregado a los módulos estándar.

- **Módulos estándar**

Los módulos estándar contienen procedimientos generales que no están asociados a ningún otro objeto y procedimientos usados con frecuencia que pueden ser ejecutados desde cualquier parte de la base de datos. Las principales diferencias entre un módulo estándar y un módulo de clase que no está asociado a un objeto determinado son el alcance y la duración. El valor de las variables y constantes declaradas o existentes en un módulo de clase sin un objeto asociado está disponible para su uso sólo cuando el código se está ejecutando y sólo desde ese objeto.

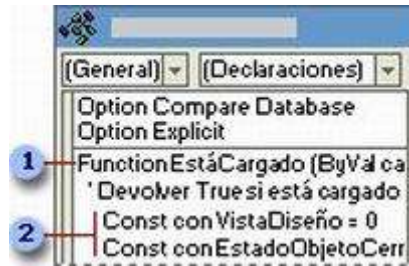


Figura 40: Módulo estándar



3. OBJETIVOS

La idea general del proyecto es el análisis, diseño e implementación de una base de datos de rutas turísticas. Dicha implementación se llevará a cabo en Access 2003, usando para ello todos los componentes que ofrece esta herramienta.

Se realizará un estudio de las características, requisitos y modelado conceptual, de una base de datos que tenga por escenario, el mapa de la provincia de Cádiz.

Se estudiarán los tipos de datos adecuados para usar con los elementos que se deben introducir en la base de datos, entre otros se introducirán, zonas, ciudades, rutas, carreteras, monumentos y alojamientos.

También se estudiarán los operadores y métodos para realizar actualizaciones de datos y mecanismos de prevención de errores, como introducción incorrecta de datos a la base de datos.

Además, se hará un estudio de los tipos de consultas a implementar, estudiando para ello los operadores y métodos más adecuados para realizar estas consultas, implementando varias de ellas sobre la base de datos generada.

Se intenta por tanto crear una base de datos con información verídica y realista, para usarla en un estudio de gestión de rutas turísticas, usando para ello formularios, intentado en todo momento que sea lo más sencillo de entender y de utilizar.

Para llevar a cabo estos objetivos se han tenido que tener una serie de conocimientos de los siguientes temas teóricos:



- 1) Conocimientos de bases de datos relacionales.
- 2) Conocimiento de Access 2003 para la inserción de datos, la realización de los formularios y consultas.
- 3) Conocimiento del lenguaje Access Basic.



4. MÉTODO DE DESARROLLO

En este apartado se va a proceder a explicar como se ha realizado el proceso de análisis, diseño e implementación de la base de datos, así como de las consultas específicas desarrolladas para la misma.

El modelo elegido para el desarrollo de la base de datos, es “el mapa de la provincia de Cáceres”, entendiéndose por éste, las rutas turísticas, ciudades, monumentos, alojamientos y carreteras que componen la provincia de Cáceres.

La información para la base de datos se ha obtenido, principalmente de las siguientes páginas Web:

- http://www.turismocaceres.org/index.php?option=com_content&task=view&id=902&Itemid=210
- <http://www.lascasasrurales.com/Provincias.asp?p=10>
- <http://www.caceresjoven.com/paginas/turismo/extremadura/ruta7-3.asp>
- <http://www.redextremadura.com/turismo/pueblos/>

También se han utilizado como medio de obtención de información libros como:

- “Por tierras de Cáceres”, Escrito por Pilar Mogollon Cano-Cortes
- “El Campo Arañuelo, Los Ibores y La Jara Cacerense”, escrito por Domingo Quijada.

Como la base de datos va a ser realizada en Access 2003, se usará como lenguaje para implementarla el Access Basic. Además, se necesitará un sistema que cuente con el sistema gestor de base de datos Access 2003.

4.1. Requisitos

A continuación se expondrán una serie de requisitos de usuario y de sistema para la correcta comprensión y ejecución de la aplicación.



4.1.1. Requisitos Hardware

- PC convencional con pantalla y teclado para poder visualizar la base de datos y para poder insertar datos por teclado.

4.1.2. Requisitos Software

- Access 2003
- Lector de Imágenes

4.1.3. Requisitos de Usuario

Para el correcto funcionamiento de la aplicación el usuario debe de poseer conocimientos a nivel usuario del funcionamiento y manejo básico de la herramienta Access 2003, para poder acceder de manera sencilla a la base de datos.

La introducción de datos en la aplicación es muy sencilla y se indicará el procedimiento paso a paso de la realización de cada una de las tareas necesarias para la creación y posteriores experimentaciones sobre la base de datos. De esta forma el usuario no necesita tener más conocimientos informáticos que los básicos para acceder al programa y escribir por teclado los datos que se le indiquen.

4.2. Dominio de la Base de Datos

En este apartado, se presentará la especificación de los requisitos que se aplican al dominio de la base de datos a crear. La idea principal del proyecto era crear una base de datos de Rutas Turísticas, para efectuar posteriormente, consultas sobre los objetos y datos de la misma.

Como escenario principal de la base de datos, se escogió el mapa de la provincia de Cáceres, con todos los elementos inherentes al mismo, es decir, rutas turísticas, ciudades, carreteras de comunicación de ciudades,



monumentos de las ciudades, tipo de turismo, alojamientos de las ciudades y comarcas en las que está dividida dicha provincia.

Para una mejor definición del escenario, se presentan a continuación los requisitos específicos que se incluirán en la base de datos:

- El mapa de Cáceres, está dividido en zonas o Comarcas. Dichas zonas deben ser almacenadas en la base de datos, con la intención de utilizar sus campos en relaciones y consultas. Para almacenar la información necesaria se ha decidido crear los siguientes campos:
 - Nombre zona. Este campo se comportará como clave principal ya que en nuestro dominio no puede haber dos zonas con el mismo nombre.
 - Provincia. En este proyecto solo se contempla la provincia de Cáceres, pero se ha creído recomendable crear este campo para futuras ampliaciones.
 - Descripción. Este campo contiene una descripción textual de la zona en cuestión.
- Además, cada zona está compuesta por ciudades, que también es necesario almacenar en la base de datos, información sobre ellas. En este caso las ciudades serán identificadas por un nombre de ciudad, ya que es imposible que varias ciudades tengan el mismo nombre dentro de una determinada provincia. Se ha elegido este campo como clave principal, por que en nuestra base de datos solo se contempla la provincia de Cáceres. Para futuras ampliaciones e incorporaciones al proyecto de varias provincia, se ha incluido un campo autonumérico "Id ciudad". Dicho campo pasaría a ser la clave principal en caso realizarse la ampliación. También se ha creado un campo "Tiempo en recorrerla", que identifica el tiempo medio que se tarda en visitar una determinada ciudad.
- A las ciudades se accede a través de carreteras. Para almacenar información sobre estas, se han creado los siguientes campos:



- Nombre de la carretera. Este campo contiene el nombre identificativo de la carretera.
- Ciudad origen. Contiene el nombre de la ciudad de donde parte la carretera.
- Ciudad destino. Contiene el nombre de la ciudad donde muere el trayecto de la carretera.
- Tipo. Contiene información sobre el tipo de carretera (Autovía, Nacional, Autopista, Comarcal...).
- Distancia. Distancia en kilómetros que tiene la carretera.
- Velocidad Media. Velocidad a que se debe ir como media durante la circulación por la carretera.
- Tiempo en recorrerla. Dicho campo es calculado, y contiene el número de horas que se tardaría en recorrer la carretera circulando a una velocidad media ($\text{Tiempo} = \text{distancia} / \text{velocidad media}$).
- Las ciudades contienen monumentos y alojamientos. Es necesario almacenar información sobre ellos ya que es importante, que a la hora de mostrar las ciudades que componen una determinada ruta turística, los visitantes sepan que monumentos deben visitar y donde pueden comer y alojarse.
- Otro de los elementos importantes a almacenar son las posibles rutas turísticas que hay en la provincia de Cáceres. Estas rutas están formadas por conjuntos de ciudades de toda la provincia, ya sean de la misma zona o no. De estas rutas se quiere almacenar:
 - Nombre Ruta. Es el nombre que identifica a una determinada ruta. En este proyecto actuará como clave principal ya que es imposible que existan varias rutas con el mismo nombre.
 - Id Ruta. Se ha incluido este campo como posible clave principal, si se desea ampliar el proyecto incluyendo más provincias.
 - Tiempo en recorrerla. Este dato es un campo calculado. Dicho campo es la suma de tiempos de recorrido de las ciudades y de las carreteras que forman una determinada ruta.



- Imagen. Este campo contiene una imagen por cada ruta almacenada en la base de datos. Dicha imagen muestra un mapa de la ruta en cuestión.

4.3. Diseño de la Base de Datos

Una vez establecido el dominio de la base de datos, se procede a realizar un modelo conceptual, el modelo Entidad/Interrelación del mismo, explicando las principales entidades y relaciones, para posteriormente realizar un modelo lógico y por último uno físico.

4.3.1. Modelo Conceptual: el modelo E/R

En el siguiente diagrama se presenta el modelo E/R:

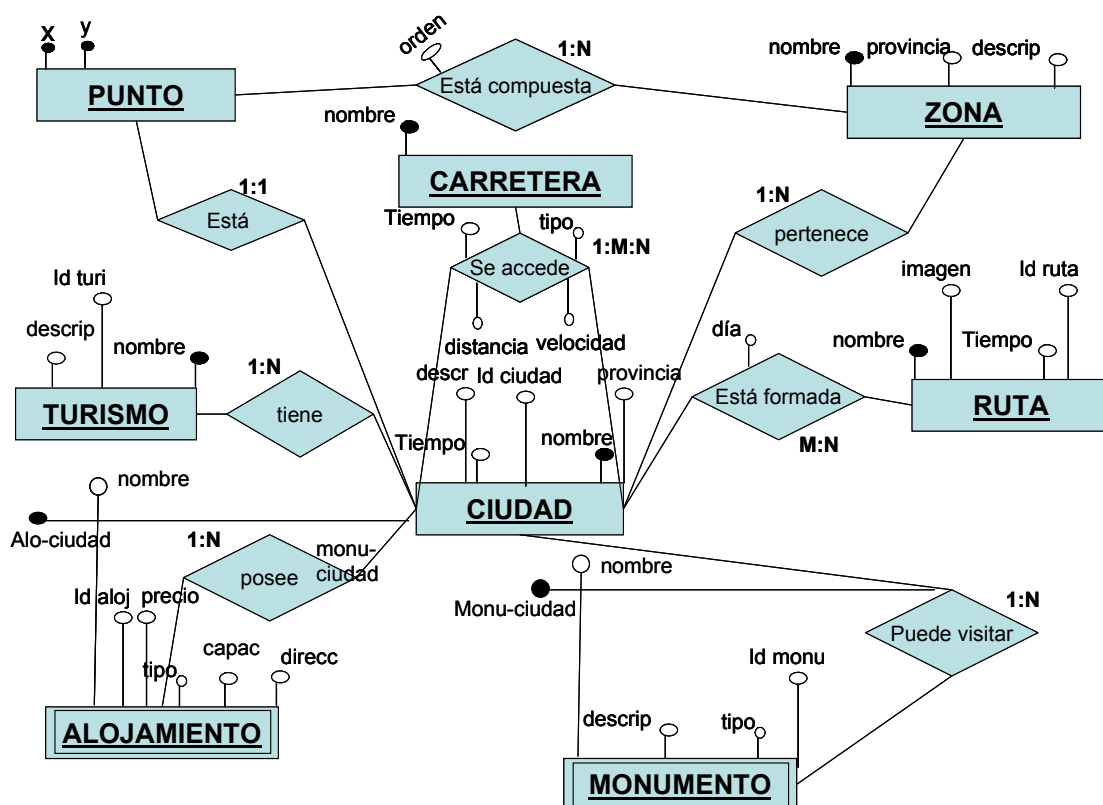


Figura 41: Diagrama E/R



La entidad principal es la de *CIUDAD* y tiene como clave primaria su nombre ya que no puede haber en una misma provincia dos ciudades con el mismo nombre. Para futuras ampliaciones (inclusión de más provincias), se ha decidido introducir un campo autonumérico (Idciudad) para que actúe como clave primaria en caso de inclusión de nuevas provincias. A parte del nombre, tiene otros atributos como:

- Tiempo de visita
- Provincia
- Descripción
- Tiempo de visita

La entidad *PUNTO* tiene como clave primaria las coordenadas X Y, en este caso siempre será un punto que indica la posición geométrica en el mapa.

La entidad *CIUDAD* tiene una relación con la entidad *PUNTO*, de tal forma, que cada *CIUDAD* *obtiene* una posición. Una *CIUDAD* puede tener 1 posiciones en el mapa, y una posición en un mapa solamente puede ser ocupada por una *CIUDAD*.

La entidad *PUNTO* también está relacionada con la entidad *ZONA*, ya que una zona esta delimitada por puntos y un punto solo puede pertenecer a una zona

En la entidad *RUTA* se puede observar su clave primaria, nombre, y sus atributos, imagen, tiempo en recorrerla (campo calculado. Es igual a la suma de tiempos de visita de las ciudades que componen la ruta y el tiempo de recorrido en carreteras de acceso a las ciudades), idruta (campo propuesto para en ser clave primaria en futuras ampliaciones). Esta entidad tiene una relación con la entidad *CIUDAD* de tal forma que una *CIUDAD* puede pertenecer ninguna o varias *RUTAS* y una *RUTA* puede estar formada por ninguna o varias *CIUDAD*.

La entidad *ZONA* se puede observar su clave primaria, nombre, y sus atributos, provincia y descripción. Esta entidad tiene una relación con la entidad *CIUDAD*



de tal forma que una *CIUDAD* puede pertenecer solo a una *ZONA* y una *ZONA* puede estar formada por 1 o varias *CIUDADES*.

La entidad *CARRETERA* tiene como clave principal el nombre de la carretera. Al no tener más atributos, a la hora de implementar dicha entidad se ha decidido eliminarla y crear solo la relación ternaria que une a la entidad *CIUDAD* con esta. Dicha relación se convierte en la implementación, en una nueva tabla con los atributos NombreCarretera (clave primaria de *CARRETERA*), CiudadOrigen (clave primaria de *CIUDAD*) y CiudadDestino (clave primaria de *CIUDAD*) como clave principal. Esta relación tiene como significado que una determinada carretera tiene una ciudad origen de donde parte y una ciudad destino donde termina. A su vez una ciudad puede ser ciudad origen o destino de varias carreteras. También tiene otros atributos como Tiempo de recorrido, distancia y velocidad media.

La entidad *MONUMENTO* tiene como clave principal el nombre del monumentos y el nombre de la ciudad a en la que se encuentra. También tiene otros atributos como descripción, dirección y tipo. Dicha entidad representa los monumentos que hay en las ciudades. Por lo tanto está relacionada con la entidad *CIUDAD* de modo que una ciudad puede tener de ninguno a varios monumentos y un monumento solo puede pertenecer a una determinada ciudad.

La entidad *ALOJAMIENTO* también tiene como clave principal el nombre del alojamiento y el nombre de la ciudad en la que se encuentra situado. A parte tiene otros atributos como tipo (si es casa rural, hotel, apartamento...), capacidad (capacidad máxima de personas que puede hospedar) y dirección (ubicación de donde está situado el alojamiento). Esta entidad está relacionada con la entidad *CIUDAD*, ya que una ciudad puede tener de ninguno a varios alojamientos, pero un alojamiento solo puede estar situado en una determinada ciudad.



4.3.2. Modelo Lógico: el modelo Relacional

```

graph TD
    ZONA["ZONA (Nombre, Provincia, Descripción)"]
    PUNTOZONA["PUNTOZONA (NombreZona, X, Y, Orden)"]
    PUNTO["PUNTO (X, Y)"]
    CIUDAD["CIUDAD (Nombre, Provincia, Descripción, NombreZona, X, Y, Idciudad)"]
    CARRETERA["CARRETERA (Nombre, CiudadOrigen, CiudadDestino, Tipo, Tiempo, Distancia, Velocidad)"]
    RUTACIUDAD["RUTACIUDAD (NombreRuta, NombreCiudad, NºdiaRecorrerla)"]
    RUTA["RUTA (Nombre, IdRuta, Tiempo, Imagen)"]
    TURISMO["TURISMO (Nombre, IdTurismo, Descripción, NombreCiudad)"]
    MONUMENTO["MONUMENTO (Nombre, IdMonumento, Tipo, Descripción, NombreCiudad, Dirección)"]
    ALOJAMIENTO["ALOJAMIENTO (Nombre, Tipo, Capacidad, IdAlojamiento, Dirección, NombreCiudad)"]

    ZONA -- "DC/UC" --> PUNTOZONA
    PUNTOZONA -- "DC/UC" --> PUNTO
    PUNTO -- "DC/UC" --> CIUDAD
    CIUDAD -- "DC/UC" --> CARRETERA
    CARRETERA -- "DC/UC" --> RUTACIUDAD
    RUTACIUDAD -- "DC/UC" --> RUTA
    RUTA -- "DC/UC" --> TURISMO
    TURISMO -- "DC/UC" --> MONUMENTO
    MONUMENTO -- "DC/UC" --> ALOJAMIENTO

```

La notación que se ha seguido para los diferentes elementos en el diagrama son: las claves primarias se han representado con una línea continua (—) debajo del nombre, las claves ajenas se han marcado con una línea continua



encima del nombre, de la cual sale una flecha que indica de que tabla procede dicho campo. Las claves que pueden tomarse como posibles claves privadas han sido marcadas con una línea discontinua (-----) debajo del nombre.

Para indicar las restricciones de borrado y actualización, se han empleado las siglas D/C para indicar borrado en cascada (Delete Cascade), D/R para indicar borrado restringido (Delete Restrict) y U/C para indicar que las actualizaciones se harán en cascada (Update Cascade).

Se ha creado una nueva tabla que proviene de la relación entre la entidad, *PUNTO* y *ZONA* ya que esta debe tener atributos propios. Esta nueva tabla se llama *PUNTOZONA* y tiene como clave principal la clave principal de *PUNTO* (XY) y la clave principal de *ZONA* (nombre de zona). A parte también tiene otro atributo como el orden de los puntos que forman la zona.

Otro de las relaciones que se ha tenido que convertir en tabla, ha sido la relación que une la entidad *RUTA* con *CIUDAD*. Dicha relación tiene cardinalidad N:M y por tanto para poder ser implementada debe tratarse como una nueva tabla a la que le ha llamado *RUTACIUDAD*. Ésta tiene como clave principal, la clave principal de la entidad *RUTA* (nombreruta) y el orden de día en el cual debe visitarse la ciudad. Esta tabla también posee un campo propio que es el nº de día en el cual se debe visitar una determinada ciudad que una ruta.

4.3.3. Modelo Físico.

Con el modelo relacional acabado, el siguiente paso a realizar es el paso a un modelo físico que pueda ser usado en el SGBD Access 2003. Para ello, se transformará el modelo relacional en un modelo físico, empleando Access Basic que es el lenguaje de definición de datos que entiende Access 2003. La estructura del modelo físico resultante será el siguiente:



RUTA

Atributo	Tipo de dato	NOT NULL	Observaciones
Nombre Ruta (PK)	Texto	✓	Nombre que recibe la ruta
Tiempo en recorrerla	Número	✓	Tiempo en recorrer la ruta
Imagen	Objeto OLE	✗	Imagen de la ruta en cuestión

CIUDAD

Atributo	Tipo de dato	NOT NULL	Observaciones
Nombre ciudad (PK)	Texto	✓	Nombre de la ciudad
provincia	Texto	✗	Provincia en la que se encuentra
Tiempo en recorrerla	Número	✓	Tipo militar/civil, avión/ultraligero/helicóptero
Nombre zona (FK)	Texto	✓	Observaciones varias
X (FK)	Número	✓	Coordenada X de su posición
Y (FK)	Número	✓	Coordenada Y de su posición
Tipo turismo	Texto	✓	Tipo de turismo que tiene la ciudad
Id_ciudad	Autonumérico	✓	Identificador numérico
Descripción	Memo	✗	Resumen de las características de las ciudades.

PUNTO

Atributo	Tipo de dato	NOT NULL	Observaciones
X (PK)	Número	✓	Coordenada x
Y (PK)	Número	✓	Coordenada y

**ZONA**

Atributo		Tipo de dato	NOT NULL	Observaciones
Nombre (PK)	Zona	Texto	✓	Nombre de la zona
Provincia		Texto	x	Nombre provincia donde se encuentra la zona
Id zona		Autonumérico	✓	Identificador numérico

PUNTOZONA

Atributo		Tipo de dato	NOT NULL	Observaciones
X(PK/FK)		Número	✓	Coordenada X de la zona en el mapa
Y (PK/FK)		Número	✓	Coordenada Y de la zona en el mapa
Nombre (PK/FK)	zona	Texto	✓	Nombre zona
Orden		Número	✓	Orden que ocupa el punto en la zona

RUTACIUDAD

Atributo		Tipo de dato	NOT NULL	Observaciones
Nombre (PK/FK)	Ruta	Texto	✓	Nombre de la ruta
Nombre (FK)	Ciudad	Texto	✓	Nombre de la ciudad
Nº día recorrerla (PK)		Número	✓	Orden del día que toca recorrer la ciudad

CARRETERA

Atributo		Tipo de dato	NOT NULL	Observaciones
Nombre (PK)	carretera	Texto	✓	Nombre común de la ayuda
Ciudad	origen	Texto	✓	Indicador OACI de la ayuda



(PK, FK)

Ciudad destino	Texto	✓	Tipo de radiobaliza, VOR, VOR/DME, TACAN, NDB, etc.
Tiempo en recorrerla	Número	✓	Tiempo que se tarda en recorrer la carretera. Es un campo calculado
Distancia	Número	✓	Distancia en km que hay desde el origen hasta el destino
Velocidad Media	Número	✓	Velocidad media a la que se puede circular por la carretera en cuestión
Tipo	Texto	✓	Tipo de carretera (A,AP, comarcal..)

MONUMENTO

Atributo	Tipo de dato	NOT NULL	Observaciones
Nombre monumento (PK)	Texto	✓	Nombre del monumento
Tipo	Texto	x	Tipo del monumento (románico, gótico...)
Descripción	Memo	x	Resumen de las características del monumento
Dirección	Memo	x	Dirección donde está situado el monumento
Nombre Ciudad (FK)	Texto	✓	Nombre de la ciudad en la que está situado el monumento
Id Monumento	Autonumérico	✓	Identificador numérico del monumento

ALOJAMIENTO

Atributo	Tipo de dato	NOT NULL	Observaciones
Nombre alojamiento (PK)	Texto	✓	Nombre del alojamiento
Tipo	Texto	x	Tipo del alojamiento (hotel, apartamento, pensión...)
Capacidad	Número	x	Nº de personas que puede hospedar el alojamiento



Dirección	Memo	x	Dirección donde está situado el alojamiento
Nombre Ciudad (FK)	Texto	✓	Nombre de la ciudad en la que está situado el alojamiento
Id Alojamiento	Autonumérico	✓	Identificador numérico del alojamiento
Precio	Moneda	✓	Precio por noche que cuesta una habitación

TURISMO

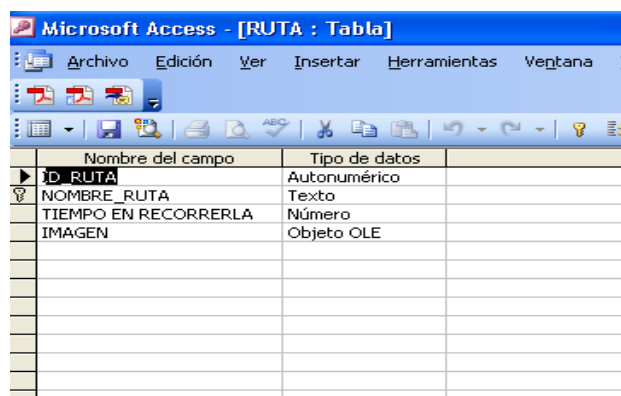
Atributo	Tipo de dato	NOT NULL	Observaciones
Nombre Turismo (PK)	Texto	✓	Nombre de turismo
Id Turismo	Autonumérico	✓	Identificador numérico del turismo
Descripción	Memo	x	Resumen de las características del turismo

Se han indicado las claves primarias poniendo a la derecha del atributo, las letras PK (Primary Key) y de la misma forma las claves ajenas se han indicado como FK (Foreign Key).

4.4. Implementación de las tablas de la Base de Datos

Las tablas que se deben implantar para crear la base de datos, son las citadas en el apartado anterior. A continuación se van a mostrar pantallazos de cómo han quedado una vez creadas en Access 2003:

- RUTA

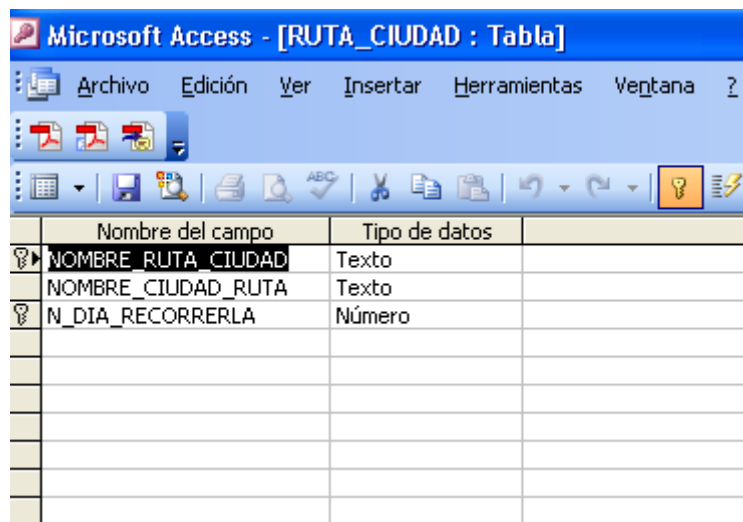


Microsoft Access - [RUTA : Tabla]		
Archivo Edición Ver Insertar Herramientas Ventana ?		
[Toolbar icons]		
Nombre del campo	Tipo de datos	
ID_RUTA	Autonumérico	
NOMBRE_RUTA	Texto	
TIEMPO EN RECORRERLA	Número	
IMAGEN	Objeto OLE	

Figura 43: Tabla Ruta



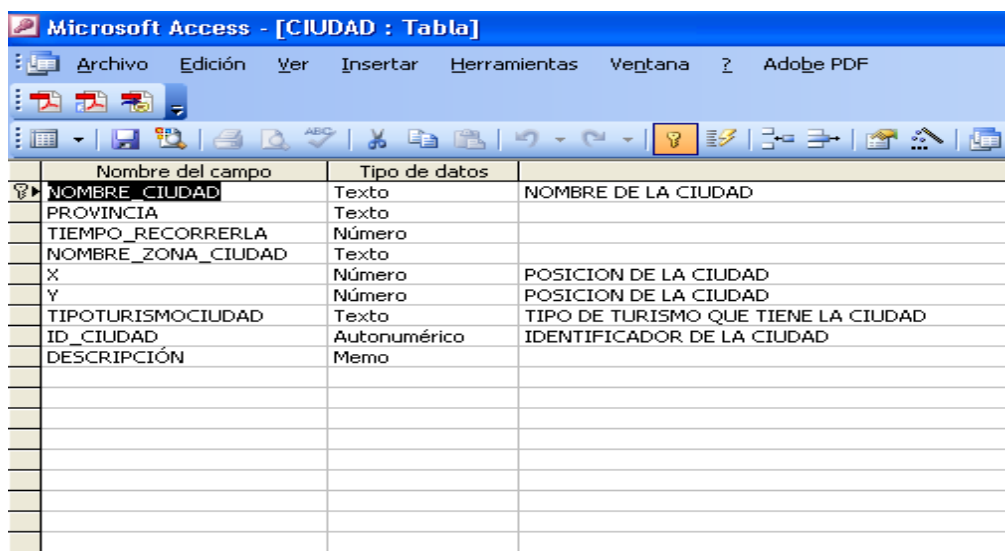
- RUTACIUDAD



Nombre del campo	Tipo de datos
NOMBRE_RUTA_CIUADAD	Texto
NOMBRE_CIUADAD_RUTA	Texto
N_DIA_RECORRERLA	Número

Figura 44: Tabla RutaCiudad

- CIUDAD

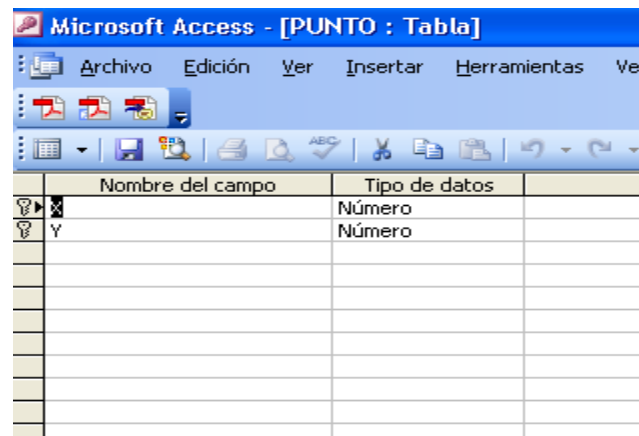


Nombre del campo	Tipo de datos
NOMBRE_CIUADAD	Texto
PROVINCIA	Texto
TIEMPO_RECORRERLA	Número
NOMBRE_ZONA_CIUADAD	Texto
X	Número
Y	Número
TIPOTURISMOCIUDAD	Texto
ID_CIUADAD	Autonumérico
DESCRIPCIÓN	Memo

Figura 45: Tabla Ciudad



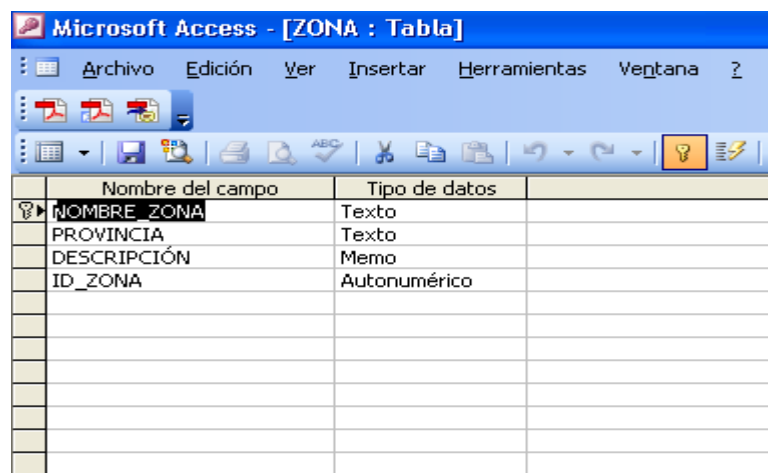
- PUNTO



Nombre del campo	Tipo de datos
ID	Número
Y	Número

Figura 46: Tabla Punto

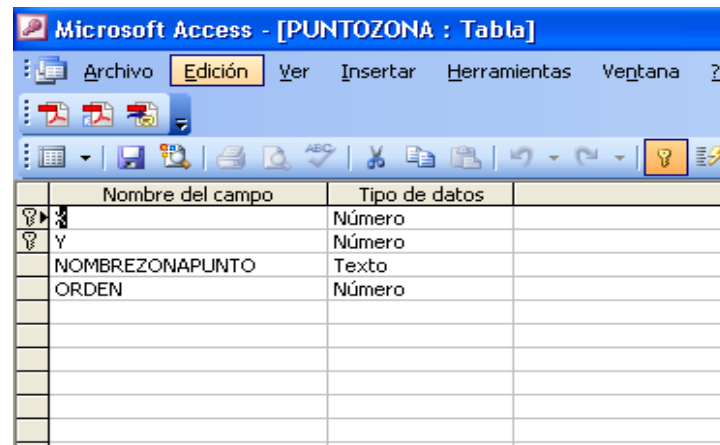
- ZONA



Nombre del campo	Tipo de datos
NOMBRE_ZONA	Texto
PROVINCIA	Texto
DESCRIPCIÓN	Memo
ID_ZONA	Autonumérico

Figura 47: Tabla Zona

- PUNTOZONA



Nombre del campo	Tipo de datos
ID	Número
Y	Número
NOMBREZONAPUNTO	Texto
ORDEN	Número

Figura 48: Tabla PuntoZona



- CARRETERA

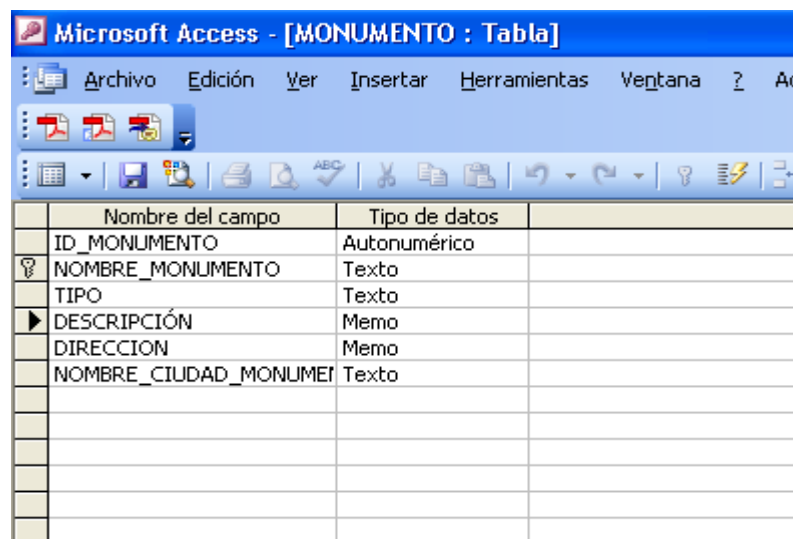


Microsoft Access - [CARRETERA : Tabla]

	Nombre del campo	Ortografía	datos
	NOMBRE_CARRETERA		Texto
	NOMBRE_CIU_ORIGEN		Texto
	NOMBRE_CIU_DESTINO		Texto
	TIPO		Texto
	TIEMPO_RECORRERLA		Número
	DISTANCIA		Número
	VELOCIDA_MEDIA		Número

Figura 49: Tabla Carretera

- MONUMENTO



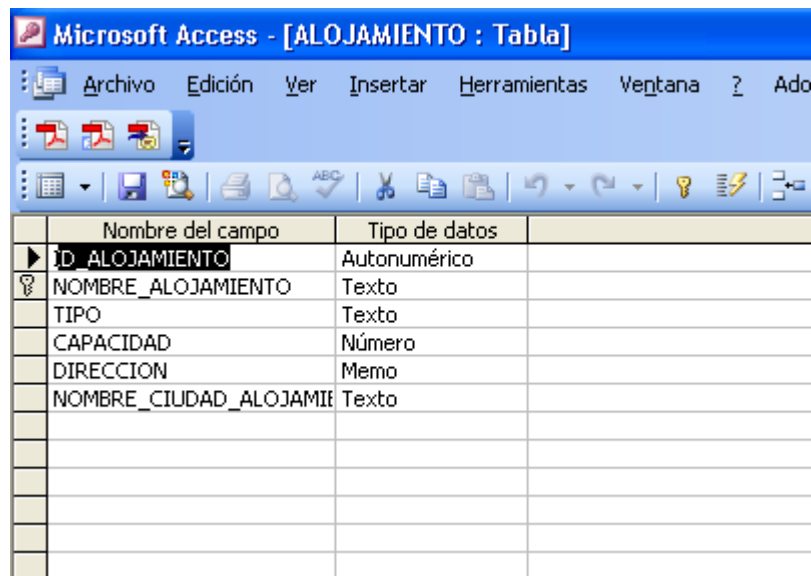
Microsoft Access - [MONUMENTO : Tabla]

	Nombre del campo	Tipo de datos
	ID_MONUMENTO	Autonumérico
	NOMBRE_MONUMENTO	Texto
	TIPO	Texto
	DESCRIPCIÓN	Memo
	DIRECCION	Memo
	NOMBRE_CIUADAD_MONUMEI	Texto

Figura 50: Tabla Monumento



- ALOJAMIENTO

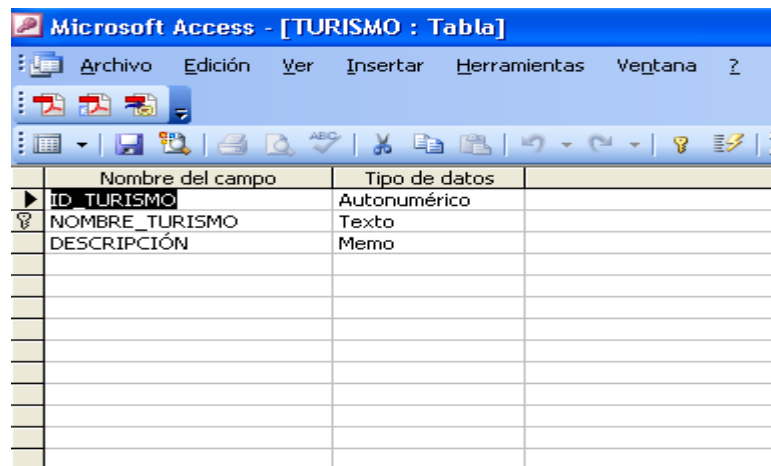


Microsoft Access - [ALOJAMIENTO : Tabla]

	Nombre del campo	Tipo de datos
►	ID_ALOJAMIENTO	Autonumérico
🔑	NOMBRE_ALOJAMIENTO	Texto
	TIPO	Texto
	CAPACIDAD	Número
	DIRECCION	Memo
	NOMBRE_CIUADAD_ALOJAMIE	Texto

Figura 51: Tabla Alojamiento

- TURISMO



Microsoft Access - [TURISMO : Tabla]

	Nombre del campo	Tipo de datos
►	ID_TURISMO	Autonumérico
🔑	NOMBRE_TURISMO	Texto
	DESCRIPCIÓN	Memo

Figura 52: Tabla Turismo

Una vez implementada la base de datos se procederá a uno de los puntos de estudio sobre la misma. El primero de estos estudios es el uso de CONSULTAS.



4.5. Relaciones

Aunque en el esquema E/R se pueden observar las relaciones que tienen los objetos de la base de datos, se ha creído conveniente mostrar una imagen de cómo han quedado una vez implementadas en Access 2003.

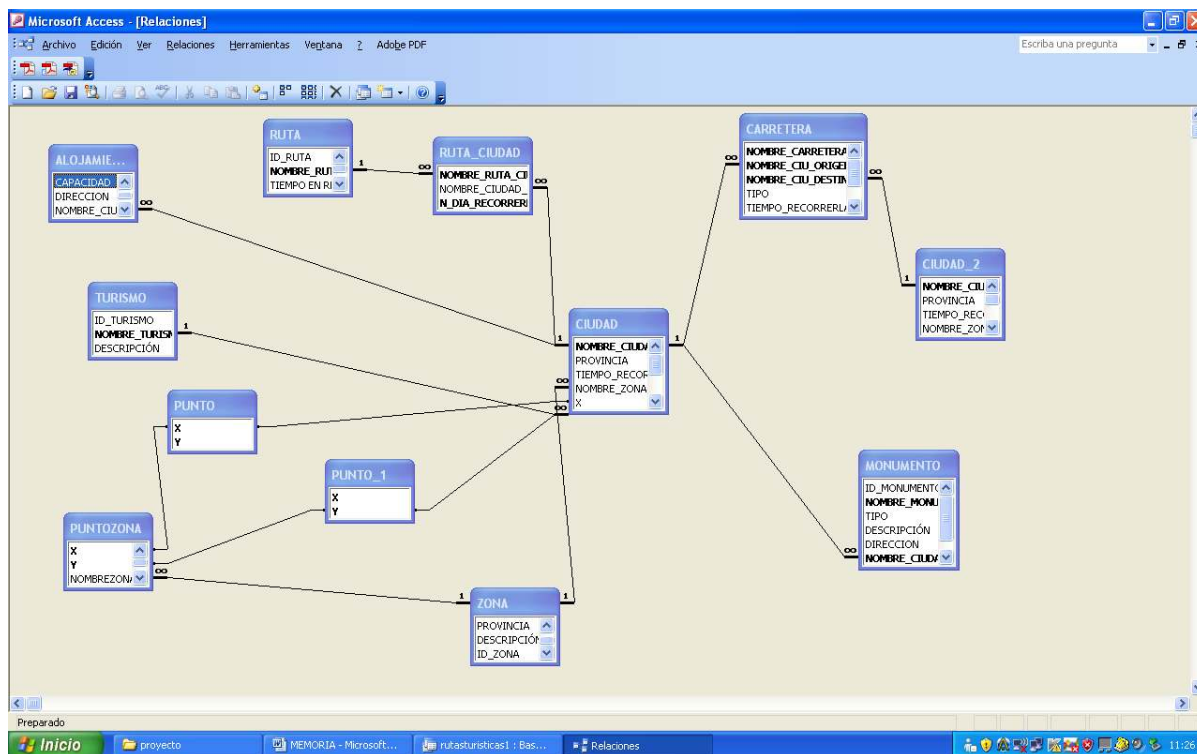


Figura 53: Esquema Relaciones

4.6. Consultas de Bases de Datos

Las consultas en bases de datos Access 2003 se pueden clasificar en varios tipos:

- Consultas de selección
- Consulta de tabla de referencias cruzadas
- Consulta de creación de tablas
- Consulta de actualización
- Consulta de datos Anexados
- Consulta de eliminación

En este proyecto tan solo se han utilizado consultas de selección.



4.6.1. Consultas de Selección

- CIUDADESPORZONA

Esta consulta muestra todas las ciudades que se encuentran dentro de una determinada ZONA. Esta zona es un dato de entrada a la consulta. Es utilizada en el formulario "RUTA_CIUADAD", el cual tiene como funcionalidad, asignar ciudades a una determinada ruta y almacenar los datos en la tabla RUTACIUADAD (dicho formulario se explicará más detalladamente en siguientes apartados).

Esta consulta devuelve el campo:

- Nombre Ciudad

CÓDIGO:

```
SELECT CIUDAD.NOMBRE_CIUADAD
FROM ZONA INNER JOIN CIUDAD ON ZONA.NOMBRE_ZONA =
CIUDAD.NOMBRE_ZONA_CIUADAD
WHERE
((( [Formularios]![RUTA_CIUADAD]![LISTAZONASCIUDAD])=[CIU
DAD]![NOMBRE_ZONA_CIUADAD]));
```

- CONSULTAVISUALIZARCIUDADESDEUNARUTA

Esta consulta devuelve las ciudades que forman una determinada ruta. Esta ruta es un parámetro de entrada a la consulta y se obtiene de un campo del formulario "pantallaconsultasRUTAS".

Esta consulta devuelve los siguientes campos:

- Orden
- Ciudad
- Zona
- Tiempo



CÓDIGO:

```
SELECT      RUTA_CIUADAD.N_DIA_RECORDERLA      AS      orden,
RUTA_CIUADAD.NOMBRE_CIUADAD_RUTA      AS      ciudad,
CIUADAD.NOMBRE_ZONA_CIUADAD      AS      zona,
CIUADAD.TIEMPO_RECORDERLA AS tiempo
FROM RUTA INNER JOIN (CIUADAD INNER JOIN RUTA_CIUADAD ON
CIUADAD.NOMBRE_CIUADAD = RUTA_CIUADAD.NOMBRE_CIUADAD_RUTA)
ON RUTA.NOMBRE_RUTA = RUTA_CIUADAD.NOMBRE_RUTA_CIUADAD
WHERE
((( [Formularios]! [CONSULTARUTA]! [Lista10]) = [RUTA_CIUADAD]
) ! [NOMBRE_RUTA_CIUADAD] And
([Formularios]! [CONSULTARUTA]! [Lista10]) = [RUTA_CIUADAD] !
[NOMBRE_RUTA_CIUADAD]) AND
(( [RUTA_CIUADAD] ! [NOMBRE_CIUADAD_RUTA] ) = [CIUADAD] ! [NOMBRE_
CIUADAD] ))
ORDER BY RUTA_CIUADAD.N_DIA_RECORDERLA;
```

- MOSTRARIMAGEN

Esta consulta muestra la imagen de una determinada ruta. Esta ruta es un parámetro de entrada a la consulta y se obtiene de un campo del formulario "pantallaconsultasRUTAS".

Esta consulta devuelve el siguiente campo:

- Imagen.

CÓDIGO:

```
SELECT RUTA.IMAGEN
FROM RUTA
WHERE
((( [Formularios]! [CONSULTARUTA]! [Lista10]) = [RUTA] ! [NOMB
RE_RUTA] )) ;
```




- OBTENER ALOJAMIENTO

Muestra los alojamientos que tiene una ciudad. Dicha ciudad es un parámetro de entrada y se consigue de un campo del formulario "consultaCIUDAD".

Esta consulta devuelve los siguientes campos:

- Nombre
- Capacidad
- Tipo
- Dirección
- Imagen
- Precio por noche

CÓDIGO:

```
SELECT                                ALOJAMIENTO.NOMBRE_ALOJAMIENTO,
ALOJAMIENTO.CAPACIDAD,                ALOJAMIENTO.TIPO,
ALOJAMIENTO.DIRECCION,                ALOJAMIENTO.IMAGEN,
ALOJAMIENTO.[PRECIO POR NOCHE]
FROM ALOJAMIENTO
WHERE
((( [ALOJAMIENTO] ! [NOMBRE_CIUADAD_ALOJAMIENTO] ) = [Formular
ios] ! [consultaCIUDAD] ! [Lista2] ))
ORDER BY ALOJAMIENTO.[PRECIO POR NOCHE];
```

- CARRETERA CON CIUDAD.

Esta consulta tiene como objetivo mostrar las ciudades que forman una ruta dada y las carreteras de acceso a ellas. La ruta se consigue del formulario "pantallaconsultasRUTAS". Para realizar esta consulta se ha realizado una llamada a la anterior consulta creada "CONSULTAVISUALIZARCIUDADESDEUNARUTA", ya que primeramente



antes de obtener las carreteras de acceso a las ciudades, es necesario averiguar los nombres de las ciudades que forman la ruta dada.

Por lo tanto los campos que devuelve son los siguientes:

- Nombre ciudad
- Nombre carretera
- Orden de visita de la ciudad en el recorrido de la ruta
- Zona
- Tiempo de visita
- Tiempo en recorrer la carretera de acceso a ella
- Distancia: kilómetros de carretera que hay que recorrer para llegar a la ciudad.
- Velocidad media: velocidad que a la que hay que circular por la carretera de acceso. Esta velocidad depende del tipo de vía del que se trate.

CÓDIGO:

```
SELECT  CONSULTAVISUALIZARCIUDADESDEUNARUTA.ciudad AS
[CIUDAD-ORIGEN],          CARRETERA.NOMBRE_CARRETERA,
CONSULTAVISUALIZARCIUDADESDEUNARUTA.orden,
CONSULTAVISUALIZARCIUDADESDEUNARUTA.zona,
CONSULTAVISUALIZARCIUDADESDEUNARUTA.tiempo,
CARRETERA.TIEMPO_RECORRERLA,          CARRETERA.DISTANCIA,
CARRETERA.[VELOCIDA_          MEDIA],
CONSULTAVISUALIZARCIUDADESDEUNARUTA.IMAGEN
FROM CONSULTAVISUALIZARCIUDADESDEUNARUTA, CARRETERA
WHERE
((( [CARRETERA]![NOMBRE_CIU_DESTINO])=[CONSULTAVISUALIZA
RCIUDADESDEUNARUTA]![ciudad]))
ORDER BY CONSULTAVISUALIZARCIUDADESDEUNARUTA.orden;
```



- OBTENER MONUMENTOS DE UNA CIUDAD

Muestra los monumentos que tiene una ciudad. Dicha ciudad es un parámetro de entrada y se consigue de un campo del formulario "consultaCIUDAD".

Esta consulta devuelve los siguientes campos:

- Ciudad
- Nombre del monumento
- Descripción
- Imagen

CÓDIGO:

```
SELECT                                MONUMENTO.NOMBRE_MONUMENTO,
MONUMENTO.DESCRIPCIÓN,                MONUMENTO.IMAGEN,
MONUMENTO.NOMBRE_CIUDAD_MONUMENTO AS CIUDAD
FROM MONUMENTO
WHERE
((( [MONUMENTO]![NOMBRE_CIUDAD_MONUMENTO])=[Formularios]
! [consultaCIUDAD]![Lista2]) AND
(( [MONUMENTO]![TIPO])=[Formularios]![OBTENERMONUMENTOCO
NSULTACIUDAD1]![Lista8])));
```

- OBTENER DATOS DE PUNTOS DE LA CIUDAD 1

Esta consulta tiene como objetivo obtener las coordenadas "X" e "Y" de una ciudad dada. La ciudad es un dato de entrada a la consulta y se obtiene del campo "Lista2", del formulario "consultaCIUDAD". Esta consulta será utilizada en otra consulta para el cálculo de distancia entre ciudades.

Por lo tanto los campos que devuelve son:

- X: coordenada "X" de la ciudad dada



- Y: coordenada "Y" de la ciudad dada

CÓDIGO:

```
SELECT CIUDAD.X AS CIUDAD1X, CIUDAD.[Y] AS CIUDAD1Y
FROM CIUDAD
WHERE
((( [CIUDAD]![NOMBRE_CIUDAD])=[Formularios]![consultaCIU
DAD]![Lista2]                                     And
([CIUDAD]![NOMBRE_CIUDAD])=[Formularios]![consultaCIUDA
D]![Lista2])));
```

- **OBTENER PUNTOS CIUDAD 2**

Esta consulta tiene como objetivo obtener las coordenadas "X" e "Y" de una ciudad dada. La ciudad es un dato de entrada a la consulta y se obtiene del campo "Lista2", del formulario "consultacalculodistanciaentreciudades". Esta consulta será utilizada en otra consulta para el cálculo de distancia entre ciudades.

Por lo tanto los campos que devuelve son:

- X: coordenada "X" de la ciudad dada
- Y: coordenada "Y" de la ciudad dada

CÓDIGO:

```
SELECT CIUDAD.X, CIUDAD.[Y]
FROM CIUDAD
WHERE
((( [CIUDAD]![NOMBRE_CIUDAD])=[Formularios]![consultacal
culodistanciaentreciudades]![Lista2])));
```



- CÁLCULO DE DISTANCIAS ENTRE DOS CIUDADES

Esta consulta tiene como objetivo calcular la distancia en kilómetros que hay entre dos ciudades dadas. La primera ciudad se obtiene del campo "Lista2" del formulario "consultaCIUDAD" y la segunda ciudad se obtiene del campo "Lista2", del formulario "consultacalculodistanciaentreciudades". Para hacer este cálculo es necesario saber las coordenadas de ambas ciudades y para ello es necesario utilizar las consultas anteriormente creadas "OBTENER DATOS DE PUNTOS DE LA CIUDAD 1", para obtener los puntos de la ciudad 1 y "OBTENER PUNTOS CIUDAD 2" para obtener los puntos de la ciudad 2. El cálculo de la distancia se ha hecho utilizando el teorema de Pitágoras ($a^2 + b^2 = c^2$).

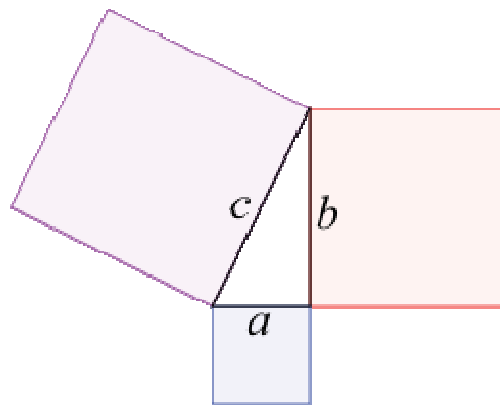


Figura 54: Esquema Teorema de Pitágoras

Por lo tanto el campo que devuelve esta consulta es:

- Distancia: distancia en línea recta entre dos ciudades.

CÓDIGO:

```
SELECT                                Sqr ( ( (OBTENERPUNTOSCIUDAD2!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) * (OBTENERPUNTO
SCIUDAD2!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) ) + ( (OBTENERPUN
TOSCIUDAD2![Y] -
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y) * (OBTENERPUNTO
```



```
SCIUDAD2! [Y] -
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y)) AS
DISTANCIA
FROM obtenerdatosdepuntosdelaCIUDAD1,
OBTENERPUNTOSCIUDAD2;
```

- OBTENER ALOJAMIENTOS DE UNA RUTA

Esta consulta tiene como función obtener los alojamientos de todas las ciudades que forman una ruta. El nombre de esta ruta, es un parámetro de entrada y se obtiene del formulario "pantallaconsultasRUTAS". Para realizar esta consulta, es necesario utilizar la consulta anteriormente creada "CONSULTAVISUALIZARCIUDADESDEUNARUTA", ya que primeramente es necesario obtener los nombres de las ciudades que forman la ruta, para posteriormente obtener sus alojamientos.

Los campos que devuelve la consulta son:

- Nombre del alojamiento
- Tipo
- Capacidad
- Ciudad
- Imagen
- Precio por noche

CÓDIGO:

```
SELECT ALOJAMIENTO.NOMBRE_ALOJAMIENTO AS NOMBRE,
ALOJAMIENTO.TIPO, ALOJAMIENTO.CAPACIDAD,
CONSULTAVISUALIZARCIUDADESDEUNARUTA.ciudad,
ALOJAMIENTO.IMAGEN, ALOJAMIENTO.[PRECIO POR NOCHE]
FROM ALOJAMIENTO, CONSULTAVISUALIZARCIUDADESDEUNARUTA
WHERE
((( [ALOJAMIENTO] ! [NOMBRE_CIUDAD_ALOJAMIENTO] ) = [CONSULTA
VISUALIZARCIUDADESDEUNARUTA] ! [ciudad] ))
ORDER BY ALOJAMIENTO.[PRECIO POR NOCHE];
```



- CONSULTA ALOJAMIENTOS POR CATERGORÍA

Esta consulta tiene como función obtener los alojamientos de una determinada categoría, de todas las ciudades que forman una ruta. El tipo de categoría es un dato de entrada a la consulta y se obtiene del campo "Lista21", del formulario "CONSULTAALOJAMIENTOPORTIPOYRUTA". Para obtener los campos a devolver se utiliza la consulta "OBTENERALOJAMIENTOSDEUNARUTA".

Por lo tanto los campos que devuelve la consulta son:

- Nombre del alojamiento
- Tipo
- Capacidad
- Ciudad
- Imagen
- Precio por noche

CÓDIGO:

```
SELECT                                OBTENERALOJAMIENTOSDEUNARUTA.NOMBRE,
OBTENERALOJAMIENTOSDEUNARUTA.[PRECIO      POR      NOCHE],
OBTENERALOJAMIENTOSDEUNARUTA.TIPO,
OBTENERALOJAMIENTOSDEUNARUTA.CAPACIDAD,
OBTENERALOJAMIENTOSDEUNARUTA.ciudad,
OBTENERALOJAMIENTOSDEUNARUTA.IMAGEN
FROM OBTENERALOJAMIENTOSDEUNARUTA
WHERE
(( (OBTENERALOJAMIENTOSDEUNARUTA.TIPO)=[Formularios]![CONSULTAALOJAMIENTOPORTIPOYRUTA]![Lista21]))
ORDER BY OBTENERALOJAMIENTOSDEUNARUTA.TIPO;
```



- CONSULTA ALOJAMIENTOS POR CATERGORÍA DE UNA CIUDAD

Esta consulta tiene como función obtener los alojamientos de una determinada categoría, de una ciudad dada. El tipo de categoría es un dato de entrada a la consulta y se obtiene del campo "Lista21", del formulario "CONSULTAALOJAMIENTODEUNACIUDADPORCATEGORIA". Para obtener los campos a devolver se utiliza la consulta "OBTENERALOJAMIENTO".

Por lo tanto los campos que devuelve la consulta son:

- Nombre del alojamiento
- Tipo
- Capacidad
- Dirección
- Imagen
- Precio por noche

CÓDIGO:

```
SELECT          OBTENERALOJAMIENTO.NOMBRE_ALOJAMIENTO,
OBTENERALOJAMIENTO.CAPACIDAD,
OBTENERALOJAMIENTO.DIRECCION,    OBTENERALOJAMIENTO.TIPO,
OBTENERALOJAMIENTO.IMAGEN,      OBTENERALOJAMIENTO.[PRECIO
POR NOCHE]
FROM OBTENERALOJAMIENTO
WHERE
(( (OBTENERALOJAMIENTO.TIPO)=[Formularios]![CONSULTAALOJ
AMIENTODEUNACIUDADPORCATEGORIA]![Lista21]));
```

- DISTANCIA MÍNIMA

Esta consulta tiene como objetivo calcular la distancia que hay entre una ciudad dada y el resto de las rutas que hay almacenadas en la base de datos. En realidad lo que hace, es calcular la distancia que hay entre todas las ciudades de la base de datos y devolver este valor junto con la ruta a la



que pertenece cada ciudad. Para realizar el cálculo se ha utilizado la consulta creada anteriormente "OBTENERDATOSDEPUNTOSDELACIUDAD1", para obtener las coordenadas de la ciudad dada. El campo "Nombre ruta", aparecerá duplicado tantas veces como ciudades tenga una determinada ruta.

Por lo tanto los campos que devuelve son:

- Distancia: distancia entre la ciudad dada y el resto de las que hay almacenadas en la base de datos
- Nombre ruta: nombre de la ruta a la que pertenecen las ciudades almacenadas en la base de datos.

CÓDIGO:

```
SELECT (Abs (Sqr ( ( (CIUDAD!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) * (CIUDAD!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) ) + ( (CIUDAD! [Y]
-obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y) * (CIUDAD! [Y]-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y) ) ) ) ) AS dmin,
RUTA_CIUADAD.NOMBRE_RUTA_CIUADAD
FROM obtenerdatosdepuntosdelaCIUDAD1, CIUDAD INNER JOIN
RUTA_CIUADAD ON CIUDAD.NOMBRE_CIUADAD =
RUTA_CIUADAD.NOMBRE_CIUADAD_RUTA
WHERE ( ( ( (Abs (Sqr ( ( ( [CIUDAD] ! [X]-
[obtenerdatosdepuntosdelaCIUDAD1] ! [CIUDAD1X] ) * ( [CIUDAD]
! [X]-
[obtenerdatosdepuntosdelaCIUDAD1] ! [CIUDAD1X] ) ) + ( ( [CIUDA
D] ! [Y]-
[obtenerdatosdepuntosdelaCIUDAD1] ! [CIUDAD1Y] ) * ( [CIUDAD]
! [Y]-
[obtenerdatosdepuntosdelaCIUDAD1] ! [CIUDAD1Y] ) ) ) ) ) ) <>0)
AND
( ( [RUTA_CIUADAD] ! [NOMBRE_CIUADAD_RUTA] ) = [NOMBRE_CIUADAD] ) )
```



```
ORDER BY (Abs (Sqr ( ( (CIUDAD!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) * (CIUDAD!X-
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1X) ) + ( (CIUDAD! [Y]
-obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y) * (CIUDAD! [Y] -
obtenerdatosdepuntosdelaCIUDAD1!CIUDAD1Y) ) ) ) ) ;
```

- PRIMER DISTANCIA MÍNIMA

La funcionalidad de esta consulta consiste en eliminar los valores duplicados del campo "Nombre Ruta", de la consulta creada anteriormente "distanciaminima", devolviendo los registros que tengan el valor mínimo en el campo "distancia minima". Para ello se ha utilizado la función "min", para calcular el valor mínimo del campo "distancia minima". Por lo tanto los campos que devuelve son los mismos que la función "distanciaminima", con la salvedad de que en este caso no muestra los valores duplicados del campo "nombre ruta".

- Distancia mínima.
- Nombre ruta. Nombre sin duplicar de la ruta que tiene la distancia mínima.

CÓDIGO:

```
SELECT      Min(distanciaminima.dmin)      AS      MínDedmin,
distanciaminima.NOMBRE_RUTA_CIUADAD
FROM distanciaminima
GROUP BY distanciaminima.NOMBRE_RUTA_CIUADAD
ORDER BY Min(distanciaminima.dmin);
```

- ELIMINAR DE LA LISTA LA RUTA A LA QUE PERTENECE LA CIUDAD

Tiene como objetivo eliminar de la consulta creada anteriormente "PRIMER DISTANCIA MINIMA", el registro que contenga en el campo "Nombre Ruta", el nombre de la ruta a la que pertenece la ciudad dada inicialmente. Para



ello primeramente es necesario obtener el nombre de la ruta a la que pertenece. Para realizar este cálculo se ha utilizado la consulta "OBTENER RUTA DE UNA CIUDAD DADA", la cual devuelve el nombre de la ruta a la que pertenece una ciudad dada.

Por lo tanto solo se devolverán los registros en los que no coincida el campo "Nombre Ruta", con el valor obtenido de la consulta "OBTENER RUTA DE UNA CIUDAD DADA".

Los campos de salida son los mismos que en la consulta "PRIMER DISTANCIA MINIMA":

- Distancia mínima.
- Nombre ruta.

CÓDIGO:

```
SELECT                                PRIMERDISTANCIAMINIMA.MínDedmin,
PRIMERDISTANCIAMINIMA.NOMBRE_RUTA_CIUADAD, RUTA.IMAGEN
FROM PRIMERDISTANCIAMINIMA, OBTENERRUTADEUNACIUADDDADA,
RUTA
WHERE
(( (PRIMERDISTANCIAMINIMA.NOMBRE_RUTA_CIUADAD) <> [OBTENERR
UTADEUNACIUADDDADA] ! [NOMBRE_RUTA_CIUADAD] ) AND
( ( [PRIMERDISTANCIAMINIMA] ! [NOMBRE_RUTA_CIUADAD] ) = [RUTA] !
[NOMBRE_RUTA] ) )
ORDER BY PRIMERDISTANCIAMINIMA.MínDedmin;
```

- OBTENER CIUDADES DE UN TIPO DE TURISMO

Tiene como objetivo, devolver todas las ciudades de la base de datos que tengan el mismo tipo de turismo que el tipo dado como entrada a la consulta. Dicho tipo se obtiene del campo "Lista2", del formulario "PANTALLACONSULTATURISMO1" y debe ser comparado con el campo "TIPOTURISMOCIUDAD", de la tabla "CIUDADA".

Los campos que devuelve son:



- Nombre ciudad
- Nombre zona
- Nombre ruta: ruta a la que pertenece la ciudad.
- Descripción.

CÓDIGO

```
SELECT CIUDAD.NOMBRE_CIUADAD, CIUDAD.NOMBRE_ZONA_CIUADAD,  
CIUDAD.DESCRIPCIÓN, RUTA_CIUADAD.NOMBRE_RUTA_CIUADAD,  
[CIUDAD]![TIPOTURISMOCIUADAD]  
FROM CIUDAD INNER JOIN RUTA_CIUADAD ON  
CIUDAD.NOMBRE_CIUADAD = RUTA_CIUADAD.NOMBRE_CIUADAD_RUTA  
WHERE  
  
((( [CIUDAD]![TIPOTURISMOCIUADAD])=[Formularios]![PANTALL  
ACONSULTATURISMO1]![Lista2]) AND  
(( [CIUDAD]![NOMBRE_CIUADAD])=[RUTA_CIUADAD]![NOMBRE_CIUAD  
D_RUTA])));
```

• OBTENER MONUMENTOS DE UNA RUTA

Esta consulta tiene como función obtener los monumentos de todas las ciudades que forman una ruta. El nombre de esta ruta, es un parámetro de entrada y se obtiene del formulario "pantallaconsultasRUTAS". Para realizar esta consulta, es necesario utilizar la consulta anteriormente creada "CONSULTAVISUALIZARCIUADADESDEUNARUTA", ya que primeramente es necesario obtener los nombres de las ciudades que forman la ruta, para posteriormente obtener sus monumentos.

Los campos que devuelve son:

- Nombre del monumento
- Tipo
- Descripción
- Nombre ciudad: nombre de la ciudad donde está situado el monumento.



- Imagen

CÓDIGO

```
SELECT      MONUMENTO.NOMBRE_MONUMENTO      AS      NOMBRE,
MONUMENTO.TIPO,
MONUMENTO.DESCRIPCIÓN,
CONSULTAVISUALIZARCIUDADESDEUNARUTA.ciudad,
MONUMENTO.IMAGEN
FROM MONUMENTO, CONSULTAVISUALIZARCIUDADESDEUNARUTA
WHERE
(( (MONUMENTO.TIPO)=[Formularios]![CONSULTAR MONUMENTOS DE
UNARUTA]![Lista17]) AND (([MONUMENTO]![NOMBRE_
CIUDAD_MONUMENTO])=[CONSULTAVISUALIZARCIUDADESDEUNARUTA
]![ciudad]));
```

- OBTENER RUTA DE UNA CIUDAD

Debe devolver como campo de salida, el nombre de las rutas a las que pertenecen todas las ciudades almacenadas en la base de datos. No tiene parámetro de entrada, tan solo debe comparar el campo "nombre" de la tabla "CIUDAD", con el campo "NOMBRE_CIUDAD_RUTA" de la tabla "RUTA_CIUDAD", para averiguar el nombre de la ruta, que en este caso es el campo "NOMBRE_RUTA_CIUDAD" de la tabla "RUTA_CIUDAD".

Los campos que devuelve son:

- Nombre ciudad.
- Nombre ruta.
- Nombre zona: zona a la cual pertenece la ciudad.

CÓDIGO:

```
SELECT      CIUDAD.NOMBRE_CIUDAD,
RUTA_CIUDAD.NOMBRE_RUTA_CIUDAD,
CIUDAD.NOMBRE_ZONA_CIUDAD
FROM      CIUDAD      INNER      JOIN      RUTA_CIUDAD      ON
CIUDAD.NOMBRE_CIUDAD = RUTA_CIUDAD.NOMBRE_CIUDAD_RUTA
```



WHERE

```
((([CIUDAD]![NOMBRE_CIU  
DAD])=[RUTA_CIU  
DAD]![NOMBRE_CIU  
DAD_RUTA]));
```

- OBTENER RUTA DE UNA CIUDAD DADA

Tiene como objetivo obtener el nombre de la ruta a la que pertenece una ciudad dada. El nombre de la ciudad es un parámetro de entrada y se consigue del campo "Lista2", del formulario "consultaCIUDAD". Para averiguar el nombre de la ruta a la que pertenece, el campo de entrada debe ser el mismo que el campo "NOMBRE_CIUAD_RUTA" de la tabla "RUTA_CIUAD".

Por lo tanto el campo que devuelve la consulta es:

- Nombre ruta.

CÓDIGO:

```
SELECT RUTA_CIUAD.NOMBRE_RUTA_CIUAD  
FROM RUTA_CIUAD  
WHERE  
(((RUTA_CIUAD)![NOMBRE_CIUAD_RUTA])=[Formularios]![c  
onsultaCIUDAD]![Lista2]));
```

- OBTENER TIPO DE TURISMO

Obtiene todos los tipos de turismo que hay almacenados en la base de datos. Para ello tan solo es necesario devolver el campo "NOMBRE_TURISMO", de la tabla "TURISMO".

Por lo tanto tan solo devuelve el campo:

- Nombre Turismo.



CÓDIGO:

```
SELECT TURISMO.NOMBRE_TURISMO  
FROM TURISMO;
```

4.7. Formularios.

Se ha creado un formulario principal llamado "home". Este formulario es el inicial, el que salta nada más arrancar la aplicación.

En este se puede observar dos enlaces diferentes:

- Insertar Datos
- Consultas

Cada uno de estos enlaces, hacen que la ejecución del programa se bifurque por ramas totalmente distintas:

- Insertar datos: lleva a los formularios específicos para insertar nuevos datos en las tablas que el usuario crea correspondientes.
- Consultas: conduce a un formulario que muestra por pantalla los datos relacionados con una ruta especificada por el usuario.

En conclusión se puede decir que el enlace "Insertar datos", conduce a formularios de inserción de datos en tablas y el enlace "consultas", conduce a formularios en los que se emplean las consultas implementadas para obtener los datos requeridos por el usuario.

El formulario "home", tiene la siguiente interfaz gráfica:



Figura 55: Formulario Home



Una vez explicado el formulario inicial, se va a proceder a explicar el resto de formularios divididos por tipos:

- **Formularios de inserción de datos**

Cuando se decide ir por los formularios de inserción de datos, el primero que aparece es uno general llamado "***PINSERTAR DATOS EN TABLA***", en el cual se puede ver enlaces a cada uno de los distintos formularios para insertar datos en tablas concretas. También se puede observar en la parte superior un enlace que permite retroceder al formulario anterior. La interfaz gráfica de este formulario es la mostrada a continuación:



Figura 56: Formulario Insertar Datos

- **Insertar datos en tabla ZONA**

Este formulario tiene como funcionalidad insertar nuevas zonas en la base de datos. Se divide en dos partes:



- Insertar datos en la tabla ZONA:
 - Nombre Zona: Insertar el nombre de la zona que se quiere insertar.
 - Provincia: Provincia a la que pertenece la zona.
 - Descripción: Breve descripción de la zona insertada
- Insertar datos de los puntos que delimitan la nueva zona en la tabla PUNTOZONA:
 - Nombre Zona Punto: Se inserta solo cuando se introduce el nombre de la nueva zona en el campo "Nombre Zona"
 - Orden: posición que ocupa el punto insertado
 - Coordenada X: coordenada X del punto insertado.
 - Coordenada Y: coordenada Y del punto insertado

Para insertar dichos puntos ha sido necesario incorporar en la interfaz un subformulario llamado "Subformulario PUNTOZONA", que ha sido creado previamente. Este es el que se encarga de introducir los nuevos datos en la tabla PUNTOZONA.

Figura 57: Formulario Insertar Datos en Zona



○ **Insertar datos en tabla CIUDAD.**

Este formulario tiene como funcionalidad insertar en la tabla CIUDAD los datos de sus correspondientes campos. Para evitar errores se ha decidido utilizar listas para mostrar todas las zonas y tipo de turismos que ya están insertados en la base de datos, tan solo el usuario debe seleccionar la que desea y se insertará de manera correcta junto con el resto de campos en el registro correspondiente de la tabla CIUDAD. Los campos de las listas se han conseguido, ejecutando las consultas implementadas anteriormente (ciudadesporzona).

Los campos que de deben insertar son:

- Nombre: nombre de la ciudad a insertar.
- Provincia: provincia a la que pertenece la ciudad
- Tiempo recorrerla: tiempo que se tarda en visitar la ciudad a insertar
- Posición: coordenadas X Y que posicionan la ciudad en el mapa
- Zona: zona en la que se encuentra la nueva ciudad
- Tipo Turismo: tipo de turismo que tiene la ciudad. En este caso una ciudad solo puede tener un tipo de turismo.

Una vez introducido los datos anteriores se debe pulsar el botón insertar para guardar la nueva ciudad en la base de datos.



The screenshot shows a web browser window with the address bar displaying 'F:\Copia Gladys\Mis documentos\proyecto\rutasturisticas1.mdb'. The page has a header with a logo, a navigation bar with 'Atrás', 'RVTA de la PLATA', and 'CIUDAD' (with a cityscape image), and a flag of Extremadura. The main content area is divided into two sections. The left section is a form titled 'Formulario Insertar Datos en Ciudad' with the following fields: 'NOMBRE:' (text input with 'ACEBO'), 'PROVINCIA:' (text input with 'CACERES'), 'TIEMPO RECORRERLA:' (text input with '1'), 'POSICIÓN:' (with 'X:' and 'Y:' sub-fields, both with '1'), 'ZONA:' (text input with 'SIERRA DE GATA'), and 'TIPO TURISMO:' (text input with 'MONTAÑA'). Below these fields is an 'INSERTAR' button. The right section is a collage of four images: a religious procession with a statue, a religious procession with a statue, a religious procession with a statue, and a religious procession with a statue.

Figura 58: Formulario Insertar Datos en Ciudad

- **Insertar datos en tabla RUTA**

Este formulario tiene como funcionalidad, insertar nuevos registros en la tabla RUTA. Para insertar un nuevo registro el usuario tan solo debe introducir el nombre de la ruta y copiar la imagen que desea que aparezca para representar dicha ruta. El tiempo total de recorrido no se introduce en este formulario ya que este es un campo calculado y se obtendrá en otro formulario de consulta.

Si lo que quiere el usuario es consultar los registros que ya están insertados en esta tabla, lo que debe hacer es ir pinchando en el incremento de número de registro situado en la parte inferior del formulario.

Figura 59: Formulario Insertar Datos en Ruta

- **Insertar datos en tabla CARRETERA**

Este formulario tiene como funcionalidad insertar los datos referentes a una nueva carretera. Igual que en casos anteriores, para evitar errores, se ha decidido crear una lista con las ciudades que hay insertadas en la base de datos. Esta lista se utiliza dos veces en el formulario, una para seleccionar la ciudad de origen y otra para seleccionar la ciudad destino. El campo tiempo recorrerla es un campo calculado que se obtiene de la división de los campos distancia y velocidad media ($\text{Tiempo} = \text{distancia} / \text{velocidad media}$). Para consultar los registros ya insertados basta con pinchar en el botón de siguiente registro situado al final del formulario.

Para insertar un nuevo registro, primero se introducen los datos y seguidamente se pulsa el botón "insertar", y este le inserta automáticamente al final de la tabla CARRETERA.



NOMBRE:

CIUDAD ORIGEN:

CIUDAD DESTINO:

TIPO:

DISTANCIA:

VELOCIDAD MEDIA:

TIEMPO RECORRERLA:

INSERTAR

ro: 1 de 11

Figura 60: Formulario Insertar Datos en Carretera

- **Insertar datos en tabla MONUMENTO**

Este formulario tiene como funcionalidad principal insertar nuevos registros en la tabla MONUMENTO. Para ello se debe introducir el nombre del monumento, el tipo, la descripción la dirección y la ciudad. Esta última, de igual forma que en el formulario anterior, para evitar errores se selecciona de una lista en la que aparecen todas las ciudades insertadas en la base de datos. Esto es importante ya al ser ese un campo referenciado de otra tabla (tabla CIUDAD), no se cae en el error de introducir un nombre que no esté en la base de datos.



Figura 61: Formulario Insertar Datos en Monumento

- **Insertar datos en tabla ALOJAMIENTO**

Este formulario tiene como funcionalidad principal insertar nuevos registros en la tabla ALOJAMIENTO. Para ello se debe introducir un nombre, tipo, capacidad de personas, una descripción de las características del monumento a insertar y la ciudad donde se encuentra el nuevo monumento. Este último campo al igual que en los formularios anteriores se debe seleccionar de una lista que contiene todas las ciudades que ya están insertadas en la base de datos. Por este motivo no se puede insertar los monumentos de una determinada ciudad si aún no se ha insertado la ciudad donde se encuentra. Para guardar los datos introducidos, por último se debe pulsar el botón "insertar".



RVTA de la PLATA

ALOJAMIENTO

Atrás

NOMBRE:

TIPO:

CAPACIDAD:

DIRECCION:

CIUDAD:

INSERTAR

EXTREMADURA

Map locations: Trigo, Plasencia, Navalmoral de la Mata, Talavera de la Reina, Caceres, Trujillo, Merida, Villanueva de la Serena, Don Benito, Badajoz, Valencia, Rio.

Figura 62: Formulario Insertar Datos en Alojamiento

- **Insertar datos en tabla TURISMO**

En este caso, este formulario tan solo tiene la funcionalidad de insertar un nuevo tipo de turismos en la tabla TURISMO. Para ello solo se debe introducir un nombre identificativo y una descripción textual de las características del nuevo tipo de Turismo. Una vez introducido todos los datos se almacena en la tabla de manera automática pulsando el botón "Insertar".



Figura 63: Formulario Insertar Datos en Turismo

- **Asignar ciudades a una determinada RUTA**

Este formulario tiene como función asignar ciudades a una determinada ruta e insertarlas en la tabla RUTA_CIUADAD. Para ello primero se selecciona de una lista, uno de los nombres de las rutas ya insertadas en la base de datos y seguidamente se le asigna un orden. Para asignar una determinada ciudad primero se hace un filtro, seleccionando una de las zonas que contiene la lista "Nombre Zona". Esta lista tiene todas las zonas que hay insertadas en el base de datos. Una vez seleccionada, aparecen en otra lista todas las ciudades que pertenecen a la zona elegida anteriormente. Por último se debe pulsar el botón "Insertar", para que automáticamente se almacenen los datos en la tabla "RUTA_CIUADAD".



Figura 64: Formulario Insertar Datos en Turismo

- **Formularios de consultas.**

Este formulario inicial de consultas, es una interfaz genérica la cual contiene enlaces a los tres tipos de consultas que se han realizado en la aplicación. Como se puede observar en la imagen, las consultas implementadas están divididas en:

- Consultas de Rutas: estas obtienen información referente a las rutas que se almacenan en la base de datos. Hay varias pero se explicarán en formularios posteriores.
- Consultas de Ciudades: estas obtienen información referente a las ciudades que se almacenan en la base de datos. De la misma manera que las consultas de Rutas, también hay varias pero se explicarán en formularios posteriores.
- Consultas Tipo de Turismo: Esta consulta tiene como función obtener las ciudades que tengan un determinado tipo de turismo



Figura 65: Formulario Consultas

- **Consultas Rutas**

Este formulario es la interfaz genérica de las consultas relacionadas con las rutas. En él se pueden encontrar enlaces a los cuatro formularios los cuales devuelven la información de la consulta realizada. Los cuatro tipos de consultas que se pueden realizar en este punto son:

- Consultar ciudades y carreteras de acceso
- Consultar alojamientos ordenados por precio
- Consultar monumentos
- Consultar alojamientos por categoría.

Cada uno de estos cuatro tipos devuelve información diferente que será explicada en los formularios posteriores.



Es importante destacar que antes de pinchar cualquier enlace, es necesario primeramente seleccionar de la lista que aparece en la parte superior del formulario, el nombre de la ruta sobre la que se quiere hacer la consulta.

Microsoft Access - [pantallaconsultasRUTAs]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana ? Adobe PDF

Escribe una pregunta

Arial 12

E:\proyecto\rutasturisticas1.mdb

RVTA de la PLATA

CONSULTAR RUTAS

SELECCIONE UNA RUTA:

RUTA CAMPO ARANUELO
RUTA DE LAS CHIMENEAS
RUTA DE LAS HURDES

CONSULTAR CIUDADES Y CARRETERAS DE ACCESO

CONSULTAR ALOJAMIENTOS ORDENADOS POR PRECIO

CONSULTAR MONUMENTOS

CONSULTAR ALOJAMIENTOS POR CATEGORÍA

Registro: 1 de 7

Vista Formulario

Inicio proyecto MEMORIA - Microsoft... rutasturisticas1 : Bas... pantallaconsultasRUTAs

12:36

Figura 66: Formulario Consultas Rutas

- **Consultar Ciudades y Carreteras de acceso**

Este formulario tiene como objetivo mostrar la información resultante de ejecutar la consulta "CARRETERA CON CIUDAD". Por lo tanto muestra las ciudades que forman la ruta seleccionada en el formulario general de "CONSULTAS RUTA", y las carreteras de acceso a estas. También muestra el tiempo estimado que se tardará en recorrer todas las ciudades de la ruta ("TIEMPO TOTAL").

Por último se puede observar que en su parte derecha muestra el mapa de ciudades que forman la ruta seleccionada. Este cambia cada vez que se selecciona una nueva ruta.



Figura 67: Formulario Consultar Ciudades y Carreteras de Acceso

○ **Consultar Alojamientos ordenados por Precio**

Este formulario muestra los alojamientos ordenados por Precio (precio por noche) que tienen todas las ciudades que forman la ruta seleccionada en el formulario "CONSULTAS RUTAS". Esta información se consigue ejecutando la consulta "OBTENER ALOJAMIENTOS DE UNA RUTA".

Como se puede ver en la imagen muestra los campos:

- Nombre
- Tipo
- Precio por noche
- Capacidad
- Ciudad en la que se encuentra.

También muestra a la derecha del formulario una foto del alojamiento en cuestión.



Microsoft Access - [CONSULTA OBTENER ALOJAMIENTOS DE UNA RUTA]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana ? Adobe PDF

Escriba una pregunta

Arial 12

E:\proyecto\rutasturisticas1.mdb

RVTA de la PLATA

ALOJAMIENTOS DE UNA RUTA

NOMBRE: HOTEL RURAL

TIPO: CASA RURAL

PRECIO POR NOCHE: 60,00 €

CAPACIDAD: 200 PERSONAS

CIUDAD: FRESNEDOSO

Registro: 2 de 2

Vista Formulario

Inicio proyecto MEMORIA - Microsoft... rutasturisticas1 : Bas... pantallaconsultasRUTAs CONSULTA OBTENER...

12:41

Figura 68: Formulario Consultar Alojamiento de una ruta ordenados por precio

○ Consultar Monumentos

Este formulario muestra los monumentos que tienen todas las ciudades que forman la ruta seleccionada en el formulario "CONSULTAS RUTAS". Para obtener la información, primeramente se debe seleccionar de la lista "TIPO", el tipo de los monumentos que se desea mostrar. Una vez hecha la selección la información solicitada se consigue ejecutando la consulta "OBTENER MONUMENTOS DE UNA RUTA". Cada vez que se cambia de tipo de monumentos, el formulario vuelve a ejecutar la consulta "OBTENER MONUMENTOS DE UNA RUTA", para actualizar los monumentos. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.

Como se puede ver en la imagen muestra los campos:

- Nombre
- Descripción
- Ciudad en la que se encuentra.



También muestra a la derecha del formulario una foto del monumento en cuestión.

Microsoft Access - [CONSULTAR MONUMENTOS DE UNA RUTA]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana ? Adobe PDF

Escriba una pregunta

Arial 10

E:\proyecto\rutasturisticas1.mdb

RVTA de la PLATA

C. MONUMENTOS DE UNA RUTA

TIPO: IGLESIA

NOMBRE: IGLESIA DE SAN PEDRO

DESCRIPCIÓN: Uno de los lugares de mayor interés turístico es la Iglesia de San Pedro Apóstol. Se inició en el siglo XV y se finalizó en el siglo XVI (1596). Mezcla formas y estructuras de gótico mudéjar con la estética del gótico final de su cabecera. Además de sus indudables valores arquitectónicos y

CIUDAD: EL GORDO

Registro: 1 de 4

Registro: 1 de 86

Vista Formulario

Inicio proyecto MEMORIA - Microsoft... rutasturisticas1 : Bas... pantallaconsultasRUTAS CONSULTAR MONUMENTOS DE UNA RUTA

Figura 69: Formulario Consultar Monumentos de una Ruta

○ Consultar Alojamientos por categoría

Este formulario consiste en mostrar los alojamientos (de las ciudades que forman la ruta seleccionada), que pertenecen a una categoría elegida por el usuario. Esta categoría debe ser pinchada de la lista "SELECCIONE UNA CATEGORÍA". Seleccionada la categoría, el formulario ejecuta la consulta "CONSULTA ALOJAMIENTO POR CATEGORÍA", que mostrará toda la información referente a los alojamientos de la ruta que pertenecen al tipo seleccionado. Esta consiste en:

- Nombre
- Tipo
- Precio por noche
- Capacidad
- Ciudad en la que se encuentra.



Cada vez que se cambia de tipo de categoría, el formulario vuelve a ejecutar la consulta "CONSULTA ALOJAMIENTO POR CATEGORÍA", para actualizar los alojamientos. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.

También muestra a la derecha del formulario una foto del alojamiento en cuestión.

Figura 70: Formulario Consultar Alojamientos de una ruta por Categoría

• Consultas Ciudades

Este formulario es la interfaz genérica de las consultas relacionadas con las ciudades. En él se pueden encontrar enlaces a los cinco formularios los cuales devuelven la información de la consulta realizada. Los cinco tipos de consultas que se pueden realizar en este punto son:

- Calcular distancia entre ciudades
- Calcular rutas más cercanas
- Consultar alojamientos ordenados por precio
- Consultar monumentos
- Consultar alojamientos por categoría.



Cada uno de estos cinco tipos devuelve información diferente que será explicada en los formularios posteriores.

Es importante destacar que antes de pinchar cualquier enlace, es necesario primeramente seleccionar de la lista que aparece en la parte superior del formulario, el nombre de la ciudad sobre la que se quiere hacer la consulta.

Figura 71: Formulario Consultas Ciudades

○ **Consultar Alojamientos ordenados por Precio**

Este formulario muestra los alojamientos ordenados por Precio (precio por noche) que tiene la ciudad que ha sido seleccionada en el formulario "CONSULTAS CIUDADES". Esta información se consigue ejecutando la consulta "OBTENER ALOJAMIENTO".

Como se puede ver en la imagen muestra los campos:

- Nombre
- Tipo
- Precio por noche



- Capacidad
- Dirección.

También muestra a la derecha del formulario una foto del alojamiento en cuestión.

Figura 72: Formulario Consultar Alojamiento de una Ciudad ordenados por precio

○ Consultar Monumentos

Este formulario muestra los monumentos que tiene la ciudad que ha sido seleccionada en el formulario "CONSULTAS CIUDADES". Para obtener la información, primeramente se debe seleccionar de la lista "TIPO", el tipo de los monumentos que se desea mostrar. Una vez hecha la selección la información solicitada se consigue ejecutando la consulta "OBTENER MONUMENTOS DE UNA CIUDAD". Cada vez que se cambia de tipo de monumentos, el formulario vuelve a ejecutar la consulta "OBTENER MONUMENTOS DE UNA CIUDAD", para actualizar los monumentos. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.



Como se puede ver en la imagen muestra los campos:

- Nombre
- Descripción

También muestra a la derecha del formulario una foto del monumento en cuestión.

Figura 73: Formulario Consultar Monumentos de una Ciudad

○ Consultar Alojamientos por Categoría

Este formulario consiste en mostrar los alojamientos de la ciudad seleccionada, que pertenecen a una categoría elegida por el usuario. Esta categoría debe ser pinchada de la lista "SELECCIONE UNA CATEGORÍA". Seleccionada la categoría, el formulario ejecuta la consulta "CONSULTA ALOJAMIENTO POR CATEGORÍA DE UNA CIUDAD", que mostrará toda la información referente a los alojamientos de la ciudad, que pertenecen al tipo seleccionado. Esta consiste en:

- Nombre
- Precio por noche
- Capacidad



- Dirección.

Cada vez que se cambia de tipo de categoría, el formulario vuelve a ejecutar la consulta "CONSULTA ALOJAMIENTO POR CATEGORÍA DE UNA CIUDAD", para actualizar los alojamientos. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.

También muestra a la derecha del formulario una foto del alojamiento en cuestión.

Figura 74: Formulario Consultar Alojamientos de una ciudad por Categoría

○ Calcular Distancia entre Ciudades

Este formulario muestra la distancia que hay entre dos ciudades seleccionadas anteriormente.

La primera ciudad, es la que se ha seleccionado en el formulario "CONSULTAS CIUDADES" y la segunda se debe seleccionar en este formulario de la lista "SELECCIONE LA CIUDAD A CONSULTAR". Una vez seleccionadas las ciudades el formulario ejecuta la consulta "CALCULO DE DISTANCIA ENTRE DOS CIUDADES". El valor que muestra el campo



"DISTANCIA" del formulario, son los kilómetros en línea recta que hay entre ambas ciudades.

Cada vez que se cambia el nombre de la segunda ciudad, el formulario vuelve a ejecutar la consulta "CALCULO DE DISTANCIA ENTRE DOS CIUDADES", para actualizar la distancia. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.

Figura 75: Formulario Calcular Distancia entre Ciudades

○ Calcular Rutas más cercanas

El formulario "CALCULAR RUTAS MÁS CERCANAS", tiene como finalidad mostrar las rutas más cercanas a una ciudad dada. Esta ciudad es la que se debe seleccionar en el formulario "CONSULTAS CIUDADES". El formulario muestra el nombre de la ruta y la distancia mínima que hay desde la ciudad a la ruta. Como se muestran varios registros estos se imprimen por pantalla de manera ascendente, es decir, el primer registro



es la ruta más cercana a la ciudad dada y el último registro es la ruta más alejada de la ciudad dada.

Figura 76: Formulario Calcular Rutas más Cercanas

• Consultas Tipo de Turismo

Este formulario muestra por pantalla la información referente a las ciudades que tienen un determinado tipo de turismo. Este tipo debe ser seleccionado por el usuario de la lista "SELECCIONE UN TIPO DE TURISMO".

La información de las ciudades se obtiene ejecutando la consulta "OBTENER CIUDADES DE UN TIPO DE TURISMO" y es mostrada por pantalla en los campos:

- Nombre Ciudad
- Zona: zona a la que pertenece la ciudad
- Ruta a la que pertenece: ruta a la que pertenece la ciudad

Cada vez que se cambia el tipo de turismo, el formulario vuelve a ejecutar la consulta "OBTENER CIUDADES DE UN TIPO DE TURISMO", para actualizar



las ciudades. Esta actualización se realiza ejecutando una macro que permite ejecutar de nuevo una consulta.

Figura 77: Formulario Consultas Tipo Turismo

5. EXPERIMENTACIÓN

5.1. CASOS DE PRUEBA DE INSERCIÓN DE DATOS

➤ Insertar datos en la tabla Zona.

Para insertar un nuevo registro de Zona es obligatorio insertar el nombre de la zona, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 78: Error no insertada clave primaria de Zona

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Zona que ya está almacenada en la base de datos. Esto se debe a que no puede haber dos zonas con el mismo nombre.



Figura 79: Error registro duplicado de Zona

Por último en este formulario se controla que no se inserten coordenadas X e Y duplicadas. Si se diese el caso de inserción de datos repetido, se mostraría un mensaje de error.

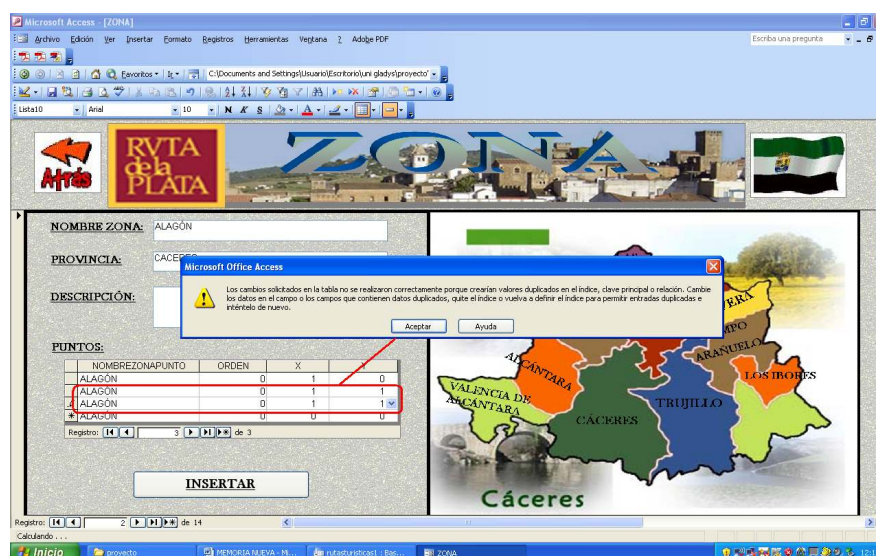


Figura 80: Error puntos duplicados de Zona

➤ Insertar datos en la tabla Ruta

Para insertar un nuevo registro de Ruta es obligatorio insertar el nombre de la Ruta, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.

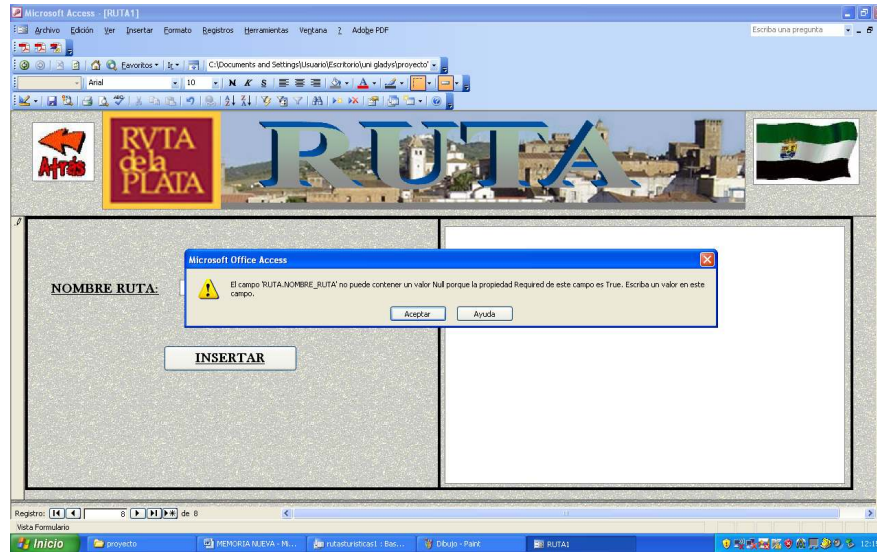


Figura 81: Error no insertada clave primaria de Ruta

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Ruta que ya está almacenada en la base de datos. Esto se debe a que no puede haber dos Rutas con el mismo nombre.

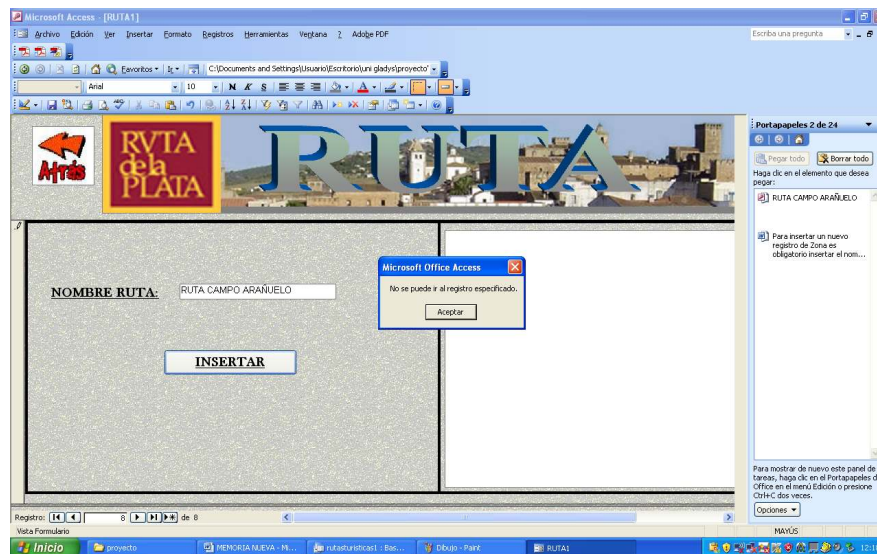


Figura 82: Error registro duplicado de Ruta

➤ Insertar datos en la tabla Ciudad

Para insertar un nuevo registro de Ciudad es obligatorio insertar el nombre de la Ciudad, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 83: Error no insertada clave primaria de Ciudad

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Ciudad que ya está almacenada en la base de datos. Esto se debe a que no puede haber dos Rutas con el mismo nombre.



Figura 84: Error registro duplicado de Ciudad

En este formulario también sale un mensaje de error si no se insertan los siguientes campos.

- Tiempo Recorrerla
- Posición: X e Y
- Zona
- Tipo Turismo



Esto se debe a que se han establecido como campos requeridos y no pueden estar vacíos.



Figura 85: Error no insertado campos obligatorios de Ciudad

➤ Insertar datos en la tabla Carretera

Para insertar un nuevo registro de Carretera es obligatorio insertar el nombre de la Carretera, el nombre de la ciudad origen y el nombre de la ciudad destino, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 86: Error no insertada clave primaria de Carretera

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Carretera, una ciudad origen y una



ciudad destino, que ya estén almacenadas en la base de datos. Esto debe a que no puede haber dos registros con la misma clave primaria.

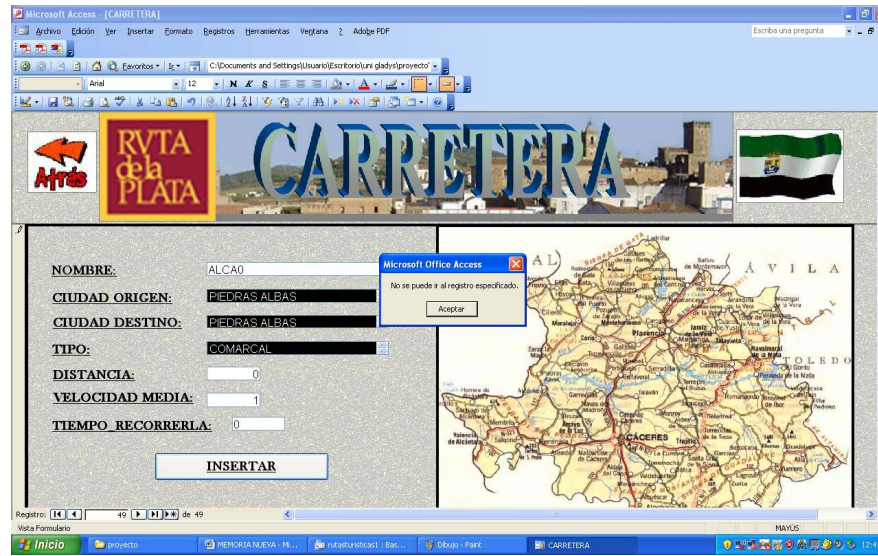


Figura 87: Error registro duplicado de Carretera

En este formulario también sale un mensaje de error si no se insertan los siguientes campos.

- Distancia
- Velocidad Media

Esto se debe a que se han establecido como campos requeridos y no pueden estar vacíos.



Figura 88: Error no insertados campos obligatorios de Carretera



➤ Insertar datos en la tabla Ciudades a Rutas

Para insertar un nuevo registro de Ciudades a Rutas es obligatorio insertar el nombre de la Ruta, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 89: Error no insertada clave primaria de Ciudad a Ruta

Lo mismo ocurre cuando no se inserta el nombre de la ciudad que se desea asignar a la ruta seleccionada. Esto se debe a que ambos campos forman la clave primaria y ninguno de ellos puede tener el valor NULL.



Figura 90: Error no insertada segunda clave primaria de Ciudad a Ruta



Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Ruta, un nombre de Ciudad y un número de orden, que ya estén almacenadas en la base de datos. Esto se debe a que no puede haber dos registros con la misma clave primaria.



Figura 91: Error registro duplicado de Ciudad a Ruta

➤ Insertar datos en la tabla Turismo

Para insertar un nuevo registro de Turismo, es obligatorio insertar el nombre del Turismo, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.

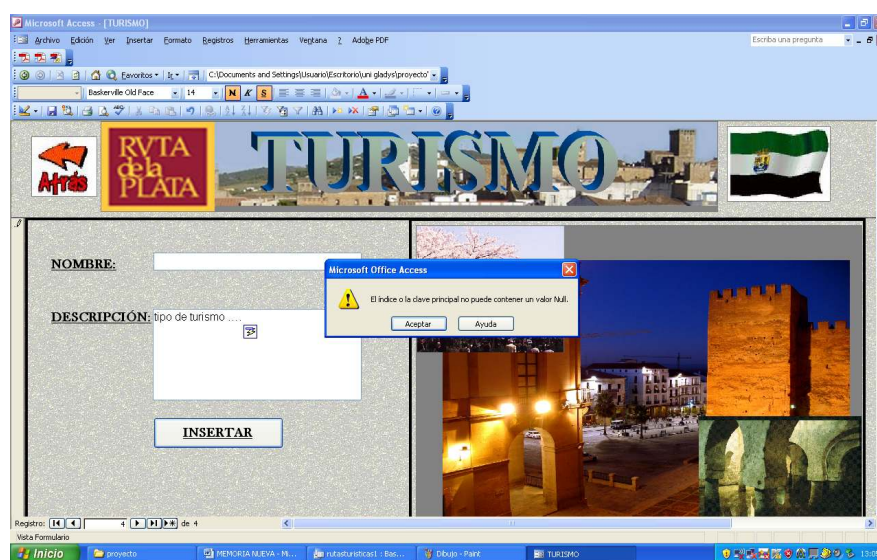


Figura 92: Error no insertada clave primaria de Turismo



Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Turismo, que ya esté almacenado en la base de datos. Esto se debe a que no puede haber dos registros con la misma clave primaria.

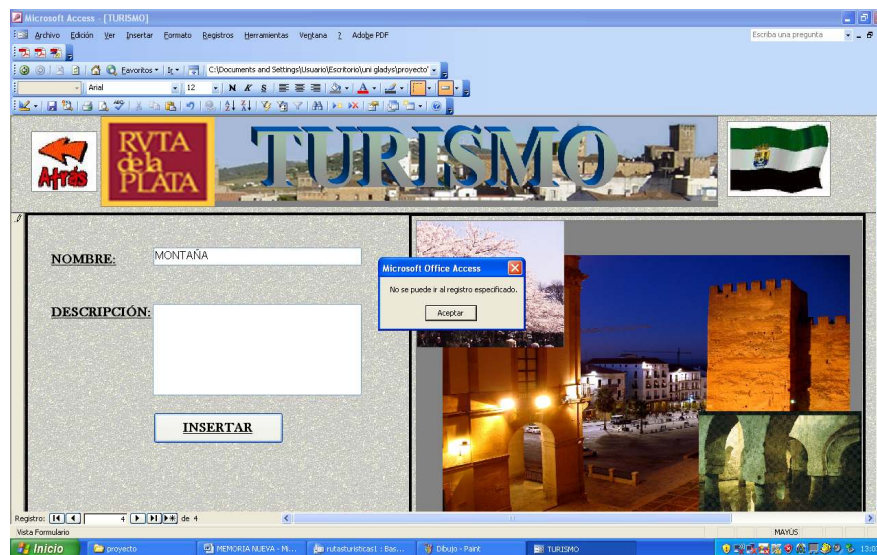


Figura 93: Error registro duplicado de Turismo

➤ Insertar datos en la tabla Monumento

Para insertar un nuevo registro de Monumento es obligatorio insertar el nombre del Monumento, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 94: Error no insertada primera clave primaria de Monumento



Lo mismo ocurre cuando no se inserta el nombre de la ciudad en la que se encuentra el monumento. Esto se debe a que ambos campos forman la clave primaria y ninguno de ellos puede tener el valor NULL.



Figura 95: Error no insertada segunda clave primaria de Monumento

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Monumento y un nombre de Ciudad, que ya están almacenados en la base de datos. Esto se debe a que no puede haber dos registros con la misma clave primaria.



Figura 96: Error registro duplicado de Monumento



➤ Insertar datos en la tabla Alojamiento

Para insertar un nuevo registro de Alojamientos es obligatorio insertar el nombre del Alojamiento, ya que de lo contrario te sale un mensaje de error y no permite insertar los nuevos datos.



Figura 97: Error no insertada primera clave primaria de Alojamiento

Lo mismo ocurre cuando no se inserta el nombre de la ciudad en la que se encuentra el alojamiento. Esto se debe a que ambos campos forman la clave primaria y ninguno de ellos puede tener el valor NULL.

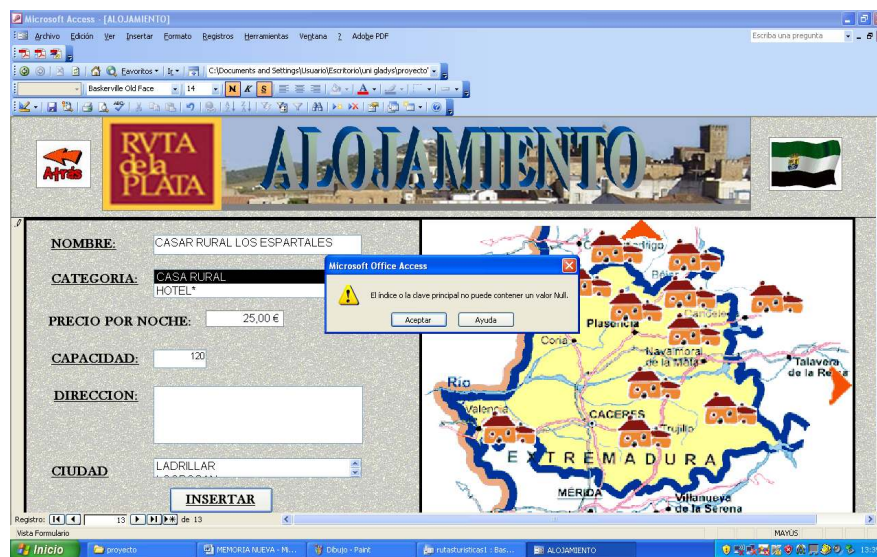


Figura 98: Error no insertada segunda clave primaria de Alojamiento

Otro caso en el cual se imprimiría por pantalla un mensaje de error, sería cuando se intenta insertar un nombre de Alojamiento y un nombre de



Ciudad, que ya están almacenados en la base de datos. Esto se debe a que no puede haber dos registros con la misma clave primaria.



Figura 99: Error registro duplicado de Alojamiento

5.2. CASOS DE PRUEBA DE CONSULTAS

- Consultas sobre Rutas
 - Consultar Ciudades y Carreteras de Acceso

Esta consulta muestra las ciudades que forman una ruta, las carreteras de acceso a ellas y el tiempo total que se tarda en recorrer la ruta. Este tiempo es la suma de todos los tiempos de visita de las ciudades más todos los tiempos que se tarda en recorrer la carretera que une una ciudad con otra.

En este ejemplo se ha seleccionado la ruta "Ruta Campo Arañuelo" y el resultado del tiempo total debería ser el siguiente:

NOMBRE_CIUADAD_RUTA	TIEMPO DE VISITA	TIEMPO EN CARRETERA
EL GORDO	1	0,375
PERALEDA DE LA MATA	4	0,416
NAVALMORAL DE LA MATA	2	0,111
MILLANES	2	0,666
BELVÍS DE MONROY	1	0,375
VALDECAÑAS DEL TAJO	2	0,444
BERROCALEJO	1	0
TOTALES	13	2,387
TIEMPO TOTAL DE RECORRIDO DE LA RUTA		$13+2,387=15,387$

Figura 100: Tabla distancias y tiempos ejemplo "Consultar Ciudades y Carreteras de Acceso"



Microsoft Access - [IMAGENRUTA]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana ? Adobe PDF

Escriba una pregunta

Inicio

proyecto

MEMORIA NUEVA - ...

rutasturisticas - B...

pantallasconsultasR...

IMAGENRUTA

Microsoft Excel - Li...

MAYUS

18:02

Figura 101: Pantalla formulario "Consultar Ciudades y Carreteras de Acceso"

Como se puede observar el resultado que da el formulario es el mismo que se ha obtenido haciendo las cuentas directamente. Por lo tanto se puede concluir que el formulario ejecuta la consulta "CARRETA CON CIUDAD", correctamente.

Para obtener el número de días se ha empleado la siguiente fórmula

$$\text{N}^{\circ}\text{días} = \text{Tiempo Total} / 8$$

Se divide entre "8", por que se considera que el día solo tiene 8 horas en las cuales se pueden visitar las ciudades.

Por lo tanto en este ejemplo el N° de días debe tener el siguiente resultado:

$$\text{N}^{\circ}\text{días} = 15,388 / 8 = 1,923$$

Se puede comprobar con la imagen, que se obtiene el mismo resultado, lo que se puede deducir que es correcto.

- Consultar Alojamientos ordenados por precio

Esta consulta muestra los alojamientos que hay en las ciudades de una ruta ordenados por precio de manera ascendente. Por ejemplo si seleccionamos la ruta "Ruta los Ibore", los alojamientos deben aparecer en el siguiente orden:



Nombre Ciudad	Nombre Alojamiento	Precio
FRESNEDOSO DE IBOR	HOTEL	50€
FRESNEDOSO DE IBOR	HOTEL RURAL	60€

Figura 102: Tabla de ejemplo Alojamientos ordenados por precio

Como se puede observar en las siguientes pantallas, el orden obtenido en el formulario es el mismo que el que aparece en la anterior tabla. Por lo tanto se puede garantizar que el formulario ejecuta la consulta "OBTENER ALOJAMIENTOS DE UNA RUTA", correctamente.

→Registro 1

Microsoft Access - [CONSULTAR ALOJAMIENTOS DE UNA RUTA]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana Ayuda PDF

Escritorio una pregunta

RVTA de la PLATA CALAJAMIENTOS DE UNA RUTA

NOMBRE: HOTEL

TIPO: HOTEL*

PRECIO POR NOCHE: 50,00 €

CAPACIDAD: 150 PERSONAS

CIUDAD: FRESNEDOSO

Registro: 14 de 2

Vista Formulario

Inicio proyecto MEMORIA NUEVA... Interactividad... pantalla consult... Microsoft Excel - la... CONSULTAR ALOJ...

Figura 102: Pantalla formulario "Consultar Alojamientos de una Ruta", registro 1

→Registro 2

Microsoft Access - [CONSULTAR ALOJAMIENTOS DE UNA RUTA]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana Ayuda PDF

Escritorio una pregunta

RVTA de la PLATA CALAJAMIENTOS DE UNA RUTA

NOMBRE: HOTEL RURAL

TIPO: CASA RURAL

PRECIO POR NOCHE: 60,00 €

CAPACIDAD: 200 PERSONAS

CIUDAD: FRESNEDOSO

Registro: 14 de 2

Vista Formulario

Inicio proyecto MEMORIA NUEVA... Interactividad... pantalla consult... Microsoft Excel - la... CONSULTAR ALOJ...

Figura 103: Pantalla formulario "Consultar Alojamientos de una Ruta", registro 2



○ Consultar monumentos

Esta consulta muestra los monumentos de un determinado tipo de la ruta seleccionada.

Por ejemplo si selecciona los monumentos del tipo "Iglesia" de la ruta "Ruta Campo Arañuelo", los registros que debe mostrar el formulario son los siguientes:

Nombre Ciudad	Nombre Monumento
El Gordo	Iglesia de San Pedro
Navalmoral de la Mata	Iglesia Santiago Apóstol
Peraleda de la Mata	Iglesia San Andrés

Figura 104: Tabla ejemplo monumentos de la ruta "Ruta Campo Arañuelo"

Como se puede observar en las siguientes pantallas, el resultado obtenido en el formulario es el correcto.

→Registro 1

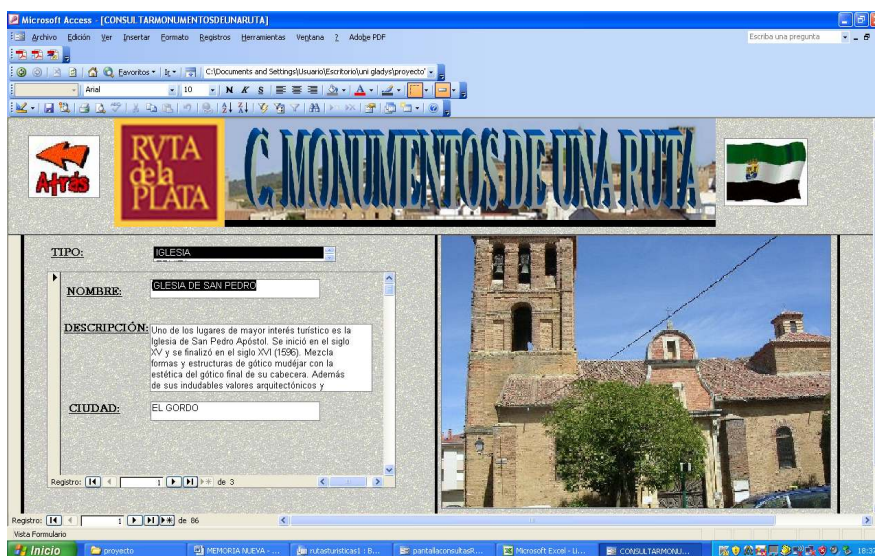


Figura 105: Pantalla formulario "Consultar Monumentos de una Ruta", registro 1



→Registro 2



Figura 106: Pantalla formulario "Consultar Monumentos de una Ruta", registro 2

→Registro 3

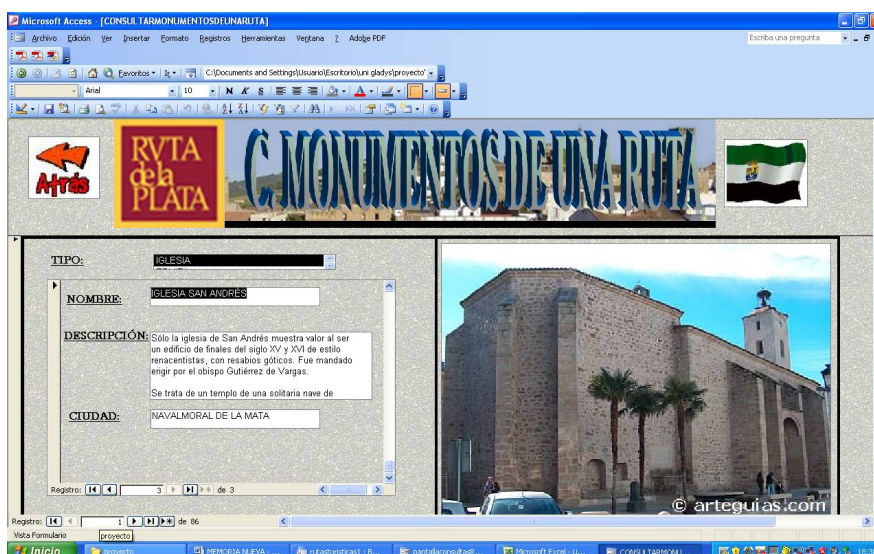


Figura 107: Pantalla formulario "Consultar Monumentos de una Ruta", registro 3

- Consultar Alojamientos por Categoría

Este formulario debe mostrar los Alojamientos de una determinada Categoría, de la ruta seleccionada. Por ejemplo si la ruta es "Ruta los Ibores", y se selecciona la categoría "Casa Rural", debe aparecer en pantalla un único registro con nombre de alojamiento "Hotel Rural". Como se observa en la siguiente imagen, el resultado es correcto.



Figura 108: Pantalla formulario "Consultar Alojamiento de una Ruta ordenados por Categoría"

➤ Consultas sobre Ciudades

○ Consultar Alojamiento ordenados por precio

Esta consulta muestra los alojamientos que hay en una determinada ciudad ordenados por precio de manera ascendente. Por ejemplo si seleccionamos la ciudad "Fresnedoso de Ibor", los alojamientos deben aparecer en el siguiente orden:

Nombre Ciudad	Nombre Alojamiento	Precio
FRESNEDOSO DE IBOR	HOTEL	50€
FRESNEDOSO DE IBOR	HOTEL RURAL	60€

Figura 109: Tabla ejemplo de Alojamiento de una Ciudad ordenados por precio

Como se puede observar en las siguientes pantallas, el orden obtenido en el formulario es el mismo que el que aparece en la anterior tabla. Por lo tanto se puede garantizar que el formulario ejecuta la consulta "OBTENER ALOJAMIENTO", correctamente.



→Registro 1

Figura 110: Pantalla formulario "Consultar Alojamientos ordenados por precio" de una ciudad, registro 1

→Registro 2

Figura 111: Pantalla formulario "Consultar Alojamientos ordenados por precio" de una ciudad, registro 2



○ Consultar Monumentos

Esta consulta muestra los monumentos de un determinado tipo de la ciudad seleccionada.

Por ejemplo si selecciona los monumentos del tipo "Ermita" de la ciudad "Peraleda de la Mata", los registros que debe mostrar el formulario son los siguientes:

Nombre Ciudad	Nombre Monumento
Peraleda de la Mata	Ermita de San Vicente
Peraleda de la Mata	Ermita del Cristo

Figura 111: Tabla ejemplo monumentos de una ciudad

Como se puede observar en las siguientes pantallas, el resultado obtenido en el formulario es el correcto.

→Registro 1

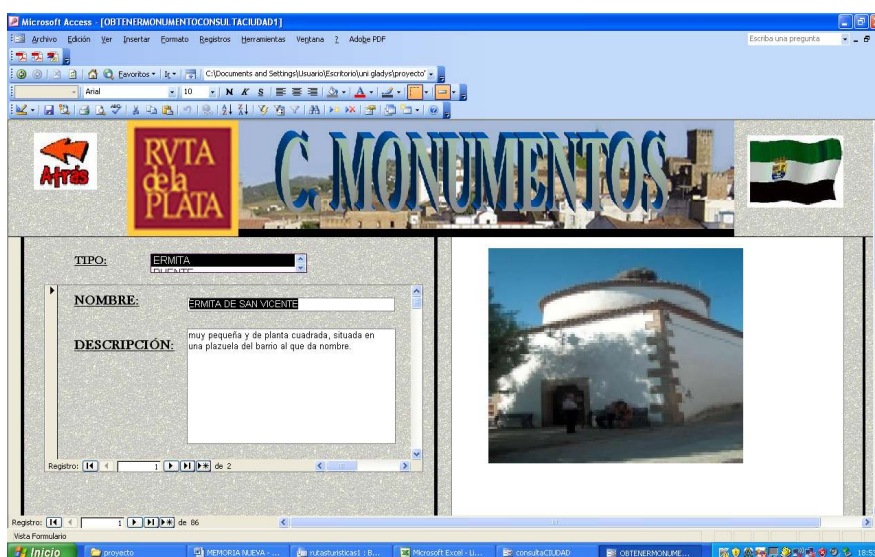


Figura 112: Pantalla formulario "Consultar Monumentos" de una ciudad, registro 1



→Registro 2

Figura 113: Pantalla formulario "Consultar Monumentos" de una ciudad, registro 2

- Consultar Alojamientos por Categoría

Este formulario debe mostrar los Alojamientos de una determinada Categoría, de la ciudad seleccionada. Por ejemplo si la ciudad es "Fresnedoso de Ibor", y se selecciona la categoría "Casa Rural", debe aparecer en pantalla un único registro con nombre de alojamiento "Hotel Rural".

Como se observa en la siguiente imagen, el resultado es correcto.

Figura 114: Pantalla formulario "Consultar Alojamientos por Categoría" de una ciudad.



○ Consultar distancia entre Ciudades

Esta consulta muestra la distancia en línea recta que hay entre dos ciudades seleccionadas. Para averiguar dicha distancia, se utiliza la fórmula del teorema de Pitágoras: $a^2 + b^2 = c^2$. Por ejemplo si seleccionamos las ciudades "Aldeanueva de la Vera" y "Peraleda de la Mata", y aplicamos la fórmula anterior, la distancia en línea recta entre ambas debería ser el siguiente:

- Aldeanueva de la Vera: X=6, Y=1
- Peraleda de la Mata: X1=2, Y1 =9

$$\text{Distancia} = |\sqrt{(X1-X)^2+(Y1-Y)^2}|$$
$$\text{Distancia} = |\sqrt{(2-6)^2+(9-1)^2}| = 8,944$$

Como se puede observar en la siguiente imagen, el formulario ejecuta la consulta "CALCULO DE DISTANCIA ENTRE DOS CIUDADES", correctamente.

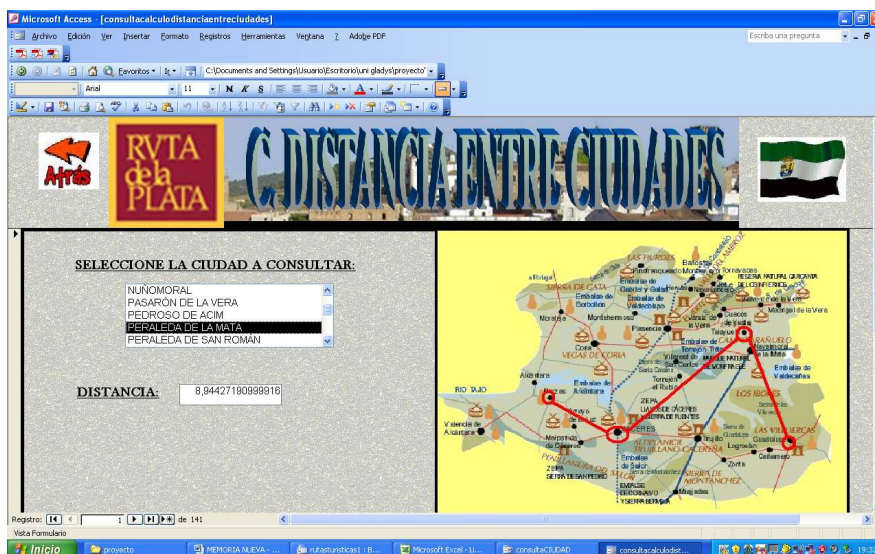


Figura 115: Pantalla formulario "Calcular distancias entre Ciudades".

○ Calcular Rutas más Cercanas

Este formulario debe mostrar por pantalla la distancia que hay desde una ciudad seleccionada a todas las rutas que están almacenadas en la base de datos. Estas rutas deben mostradas de forma ascendente, en función a la distancia calculada. Por ejemplo si se selecciona la ciudad "Berrocalejo", el formulario debe devolver los siguientes datos y en el siguiente orden:

Nombre Ruta	Distancia
ruta de las chimeneas	1
ruta los dolmenes	1,414



RUTA DE LAS HURDES	4
RUTA DE LOS IBORES	6
RUTA SIERRA DE GATA	7

Figura 116: Tabla ejemplo de rutas más cercanas.

La distancia se ha calculado empleando la fórmula explicada en el apartado anterior (teorema de Pitágoras). Para ello se ha calculado la distancia entre la ciudad seleccionada y todas las almacenadas en la base de datos. Una vez realizado el cálculo solo se muestra la distancia mínima de cada ruta.

Como se puede observar en las siguientes imágenes, el formulario muestra los datos correctamente.

→Registro 1



Figura 117: Pantalla formulario "Calcular rutas más cercanas", registro 1

→Registro 2

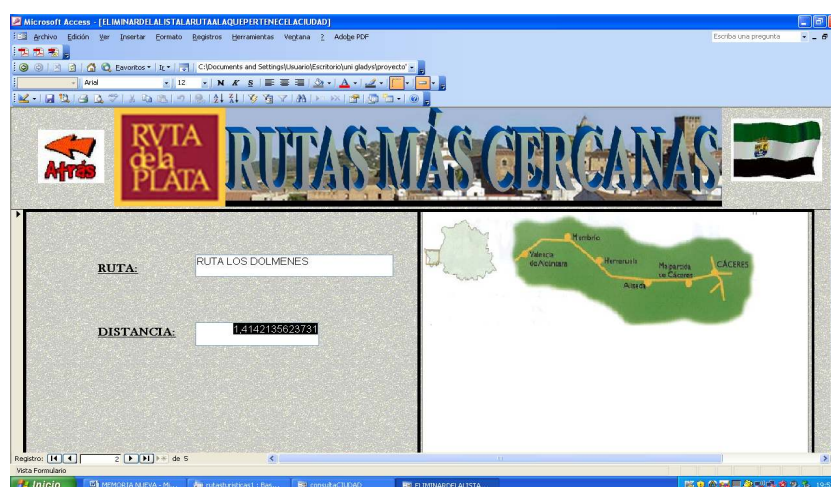


Figura 118: Pantalla formulario "Calcular rutas más cercanas", registro 2



→Registro 3

Figura 119: Pantalla formulario "Calcular rutas más cercanas", registro 3

→Registro 4

Figura 120: Pantalla formulario "Calcular rutas más cercanas", registro 4



→Registro 5



Figura 121: Pantalla formulario "Calcular rutas más cercanas", registro 5

➤ Consultas Tipo de Turismo

Esta consulta debe mostrar las ciudades que tienen un determinado tipo de turismo. Por ejemplo si se selecciona como tipo de turismo "Montaña", deben aparecer los siguientes registros:

Tipo de Turismo	Nombre Ciudad
Montaña	EL GORDO
Montaña	PERALEDA DE LA MATA
Montaña	HERRERUELA
Montaña	MALPARTIDA DE CACERES
Montaña	ALCÁNTARA
Montaña	MATA DE ALCÁNTARA
Montaña	NAVALMORAL DE LA MATA
Montaña	MALPARTIDA DE CACERES
Montaña	BERROCALEJO

Figura 121: Tabla ejemplo de ciudades de un tipo de turismo

Como se puede observar en las siguientes imágenes, el formulario muestra los registros correctos:



→Registro 1

Figura 122: Pantalla formulario "Consulta Turismo", registro 1

→Registro 2

Figura 123: Pantalla formulario "Consulta Turismo", registro 2



→Registro 3

Microsoft Access [PANTALLA CONSULTA TURISMO1]
Archivo Edición Ver Insertar Formato Registros Herramientas Ventana Ayuda PDF
C:\Documents and Settings\Usuario\Escritorio\proyecto\proyecto
And 12
Escriba una pregunta

SELECCIONE UN TIPO DE TURISMO:
MONTANA
PLAYA
TURISMOS RURAL

NOMBRE CIUDAD: HERRERUELA
ZONA: CACERES
RUTA A LA QUE PERTENECE: RUTA LOS DOLMENES

Registro: 14 de 9

Inicio proyecto MEMORIA NUEVA - N... ruta turística1 - Bas... Microsoft Excel - Libro1 PANTALLA CONSULTA...

Figura 124: Pantalla formulario "Consulta Turismo", registro 3

→Registro 4

Microsoft Access [PANTALLA CONSULTA TURISMO1]
Archivo Edición Ver Insertar Formato Registros Herramientas Ventana Ayuda PDF
C:\Documents and Settings\Usuario\Escritorio\proyecto\proyecto
And 12
Escriba una pregunta

SELECCIONE UN TIPO DE TURISMO:
MONTANA
PLAYA
TURISMOS RURAL

NOMBRE CIUDAD: MALPARTIDA DE CACERES
ZONA: CACERES
RUTA A LA QUE PERTENECE: RUTA LOS DOLMENES

Registro: 14 de 9

Inicio proyecto MEMORIA NUEVA - N... ruta turística1 - Bas... Microsoft Excel - Libro1 PANTALLA CONSULTA...

Figura 125: Pantalla formulario "Consulta Turismo", registro 4



→Registro 5

Figura 126: Pantalla formulario "Consulta Turismo", registro 5

→Registro 6

Figura 127: Pantalla formulario "Consulta Turismo", registro 6



→Registro 7

Figura 128: Pantalla formulario "Consulta Turismo", registro 7

→Registro 8

Figura 129: Pantalla formulario "Consulta Turismo", registro 8



→Registro 9

Figura 130: Pantalla formulario "Consulta Turismo", registro 9



6. CONCLUSIONES

Una vez desarrollado el proyecto y cotejado el resultado con los objetivos expuestos al principio del mismo, se ha llegado a la conclusión de que se han conseguido llevar a cabo todos ellos, de una manera bastante correcta.

Al comienzo del proyecto, no se tenían apenas conocimientos sobre los objetivos y sobre lo que se quería conseguir con la bases de datos a implementar, tampoco se conocía en profundidad la herramienta utilizar (access 2003), ya que su utilización anteriormente solo había sido para realizar tareas básicas y sencillas. Además no se tenía la información que contendría la base de datos y no había ninguna base de datos con información previa disponible. Todo esto presentaba complicado el desarrollo de este proyecto.

Las lecturas de libros, manuales y documentos ayudaron a comprender la problemática e importancia de las bases de datos. Además, contribuyeron a tener una idea general sobre la estructura y pasos a realizar.

Sin duda, desde el primer modelo E/R que se desarrollo hasta el último incluido en el proyecto, se han producido muchos cambios de tal forma que sólo se parecen en unas cuantas entidades. Esto demuestra que el proyecto se ha ido desarrollando progresivamente y adaptándose en cada paso a los diferentes cambios, debidos a nuevas características ofrecidas por las herramientas o a nuevos documentos con información relevante sobre el tema.

La base de datos se fue desarrollando poco a poco, transformando datos en los tipos adecuados para su introducción en la misma, calculando coordenadas de las ciudades para situarlas en las zonas correctas, y comprobando el correcto funcionamiento de disparadores para verificar que el sistema funcionaba correctamente.



La experimentación con la inserción de datos y consultas ha sido muy interesante y ha servido para comprender mucho mejor los aspectos de la base de datos implementada y de los errores que esta debe tener en cuenta y no permitirlos.

Se puede resolver que los objetivos iniciales por tanto, se han cumplido de forma aceptable y se han obtenido unos resultados bastantes satisfactorios.

Ha sido muy satisfactorio trabajar en un proyecto tan interesante, sobre una materia con tan actual y con tantas cosas que desarrollar.



7. DESARROLLOS POSTERIORES

Para futuras ampliaciones de este proyecto se puede establecer como nuevos requisitos los siguientes:

1. Ampliar el número de provincias y ciudades españolas para poder realizar las consultas implementadas de una manera más amplia.
2. Implementar nuevas tablas de la base de datos para que pueda guardar más información. Por ejemplo sería recomendado insertar las siguientes tablas:
 - a. Una tabla de "Actividades de Ocio" que se pueden hacer en las distintas ciudades que se almacenan en la base de datos.
 - b. Una tabla de datos para almacenar las fiestas populares de las ciudades.
 - c. Una tabla para guardar información sobre los restaurantes que hay en las distintas ciudades de la base de datos.
 - d. Una tabla para guardar información sobre los tipos de transportes públicos con los que se puede viajar a las ciudades.
 - e. Una tabla para guardar información sobre los tipos de transportes públicos que tienen las ciudades para moverse por el interior de las ciudades.
3. Implementar una nueva funcionalidad para poder reservar y comprar billetes de transporte públicos para viajar de una ciudad a otra. Para ello se debe crear una nueva tabla para almacenar información referente a los datos de las personas y de los viajes que se desean comprar o reservar.



4. Implementar una nueva funcionalidad para que el sistema te muestre el recorrido que se debe hacer para ir desde una ciudad origen hasta una ciudad destino.
5. Implementar nuevas consultas de la base de datos para obtener más información relevante. Las consultas que se pueden crear son las siguientes:
 - a. Consulta para obtener información referente a los datos de los viajes que hay disponibles para viajar en transporte público de una ciudad a otra.
 - b. Consulta que grave las reservas y compras de billetes en la base de datos.
 - c. Consulta que muestre las actividades que se pueden realizar en cada una de las ciudades que seleccione el usuario.
 - d. Consulta que muestre las actividades que se pueden realizar en cada una de las rutas que seleccione el usuario.
 - e. Consulta que muestre los restaurantes que hay en cada una de las ciudades que seleccione el usuario.
 - f. Consulta que muestre los restaurantes que hay en cada una de las rutas que seleccione el usuario.
 - g. Consulta que muestre las fiestas populares que hay en cada una de las ciudades que seleccione el usuario.
 - h. Consulta que muestre las fiestas populares que hay en las ciudades de alrededor de una ciudad seleccionada por el usuario.
6. Un punto importante sería hacer nuevos formularios que permitiese insertar los datos en las nuevas tablas creadas y visualizar los datos de las consultas que se deben implementar.
7. Por último sería recomendado implementar un nuevo formulario de ayuda que permita resolver al usuario todas las dudas de sobre como utilizar el sistema.



8. BIBLIOGRAFIA

Para la realización de las distintas partes de este documento y del proyecto han sido consultados los siguientes ejemplares, artículos y direcciones de Internet:

- <http://mit.ocw.universia.net/curso11208/11/11.208/IAP02/lecture-notes/lecture5-2.html>
- <http://www.monografias.com/trabajos11/basda/basda.shtml>
- http://www.turismocaceres.org/index.php?option=com_content&task=view&id=902&Itemid=210
- <http://www.lascasasrurales.com/Provincias.asp?p=10>
- <http://www.caceresjoven.com/paginas/turismo/extremadura/ruta7-3.asp>
- <http://www.redextremadura.com/turismo/pueblos/>
- <http://www.mailxmail.com/curso-introduccion-access/formularios-1>
- <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/manualdeaccess/formularios.htm>
- “Por tierras de Cáceres”, Escrito por Pilar Mogollon Cano-Cortes
- El Campo Arañuelo, Los Ibores y La Jara Cacerense”, escrito por Domingo Quijada.

